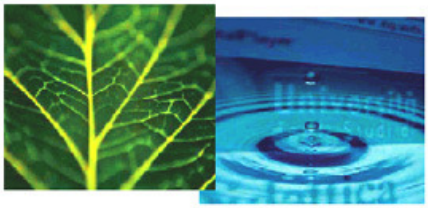
PhD Dissertation



International Doctorate School in Information and Communication Technologies

DISI - University of Trento

Privacy-Aware Risk-Based Access Control Systems

Nadia Metoui

Advisor:

Prof. Alessandro Armando

University of Genova

Co-advisor:

Dr. Michele Bezzi

SAP Security Research

Examination committee:

Prof. Pierangela Samarati - University of Milan

Prof. Stefano Paraboschi - University of Bergamo

Prof. Alessio Merlo - University of Genova

Submission: January 31^{st}, 2018 - Defense: May 4^{th}, 2018

إن صرخت بكل قواك، ورد عليك الصدى

"مَنْ هناك؟" فقل للهوية: شكراً!

محمود درويش

To mum and dad,
To my sister,
To my brother,
To friends,
To family,
To childhood dreams,
To Home, To Tounes.

Abstract

Modern organizations collect massive amounts of data, both internally (from their employees and processes) and externally (from customers, suppliers, partners). The increasing availability of these large datasets was made possible thanks to the increasing storage and processing capability. Therefore, from a technical perspective, organizations are now in a position to exploit these diverse datasets to create new data-driven businesses or optimizing existing processes (real-time customization, predictive analytics, etc.). However, this kind of data often contains very sensitive information that, if leaked or misused, can lead to privacy violations.

Privacy is becoming increasingly relevant for organization and businesses, due to strong regulatory frameworks (e.g., the EU General Data Protection Regulation GDPR, the Health Insurance Portability and Accountability Act HIPAA) and the increasing awareness of citizens about personal data issues. Privacy breaches and failure to meet privacy requirements can have a tremendous impact on companies (e.g., reputation loss, non-compliance fines, legal actions). Privacy violation threats are not exclusively caused by external actors gaining access due to security gaps. Privacy breaches can also be originated by internal actors, sometimes even by trusted and authorized ones. As a consequence, most organizations prefer to strongly limit (even internally) the sharing and dissemination of data, thereby making most of the information unavailable to decision-makers, and thus preventing the organization from fully exploit the power of these new data sources.

In order to unlock this potential, while controlling the privacy risk, it is necessary to develop novel data sharing and access control mechanisms able to support risk-based decision making and weigh the advantages of information against privacy considerations. To achieve this, access control decisions must be based on an (dynamically assessed) estimation of expected cost and benefits compared to the risk, and not (as in traditional access control systems) on a predefined policy that statically defines what accesses are allowed and denied.

*In Risk-based access control for each access request, the corresponding risk is estimated and if the risk is lower than a given threshold (possibly related to the trustworthiness of the requester), then access is granted or denied. The aim is to be more permissive than in traditional access control systems by allowing for a better exploitation of data. Although existing risk-based access control models provide an important step towards a better management and exploitation of data, they have a number of drawbacks which limit their effectiveness. In particular, most of the existing risk-based systems only support binary access decisions: the outcome is "**allowed**" or "**denied**", whereas in real life we often have exceptions based on additional conditions (e.g.,* "I cannot provide this information, unless you sign the following non-disclosure agreement." *or* "I cannot disclose this data,

because they contain personal identifiable information, but I can disclose an anonymized version of the data.*"). In other words, the system should be able to propose risk mitigation measures to reduce the risk (e.g., disclose partial or anonymized version of the requested data) instead of denying risky access requests. Alternatively, it should be able to propose appropriate trust enhancement measures (e.g., stronger authentication), and once they are accepted/fulfilled by the requester, more information can be shared.*

The aim of this thesis is to propose and validate a novel privacy enhancing access control approach offering adaptive and fine-grained access control for sensitive data-sets. This approach enhances access to data, but it also mitigates privacy threats originated by authorized internal actors. More in detail:

1. *We demonstrate the relevance and evaluate the impact of authorized actors threats. To this aim, we developed a privacy threats identification methodology EPIC (**E**valuating **P**rivacy violation r**I**sk in **C**yber security systems) and apply EPIC in a cybersecurity use case where very sensitive information is used.*

2. *We present the privacy-aware risk-based access control framework that supports access control in dynamic contexts through trust enhancement mechanisms and privacy risk mitigation strategies. This allows us to strike a balance between the privacy risk and the trustworthiness of the data request. If the privacy risk is too large compared to the trust level, then the framework can identify adaptive strategies that can decrease the privacy risk (e.g., by removing/obfuscating part of the data through anonymization) and/or increase the trust level (e.g., by asking for additional obligations to the requester).*

3. *We show how the privacy-aware risk-based approach can be integrated to existing access control models such as RBAC and ABAC and that it can be realized using a declarative policy language with a number of advantages including usability, flexibility, and scalability.*

4. *We evaluate our approach using several industrial relevant use cases, elaborated to meet the requirements of the industrial partner (SAP) of this industrial doctorate.*

Keywords
Risk; Trust; Privacy; Privacy threat assessment; Access control; Privacy enhancing technologies; Anonymization

Acknowledgements

Firstly, I would like to express my sincere gratitude to my advisors Prof. Alessandro Armando and Dr. Michele Bezzi for their continuous support of my Ph.D. study and related research. Your guidance and advice were determinant for the accomplishment of the work presented in this thesis. It has been a privilege and an honor to work with you.

My sincere thanks also go to Prof. Claudio Bettini and Dr.Sergio Mascetti with whom I had the great chance to collaborate during the final stages of this Ph.D., and from whom I learned a lot.

I would like to thank Prof. Pierangela Samarati (University of Milan, Italy), Prof. Stefano Paraboschi (University of Bergamo, Italy), and Prof. Alessio Merlo (University of Genova, Italy), who granted me the honour of accepting to evaluate my Ph.D. as examination committee.

I would like to thank my friends and fellow ESRs in the SECENTIS project Stanislav Dashevskyi, Daniel Ricardo Dos Santos, Mojtaba Eskandari, and Avinash Sudhodanan. Shared hardships and shared joys forged the best of camaraderies.

I would also like to thank all the wonderful people I met at SAP Labs France, Fondazione Bruno Kessler and Studentato San Bartolameo. I will always remember all the fun (and some of the serious) moments we shared.

Last but not the least, I would like to thank my family for all their love and encouragement throughout all my pursuits.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Data, including personal information, is an increasingly valuable asset for modern businesses and organizations. Indeed the amount of data collected by 2020 is expected to exceed 44 billion of gigabytes worldwide [71] and the European Commission estimates the European data market value to reach € 739 billion [72] by the same year. If carefully handled this data can enable organizations to understand and react with precision to customers and stakeholders needs, providing tangible competitive advantages in the marketplace.

However, the sensitive and personal nature of data is also increasing the burden for companies, which are subject to strict regulation for collection, processing and sharing data. In addition to the possible fines and sanctions prescribed by data protection laws (e.g., in the Europe fines can reach 20 million euros or 4% of the global annual turnover for companies failing to meet the EU General Data Protection Regulation GDPR requirements [57]), privacy breaches can have also a huge impact on companies reputation and relationships with partners, clients, and employees, which has the potential to dramatically increase the bill.

Therefore, companies must conduct thoroughly privacy threat investigations and evaluation to identify and remediate to any gaps with respect to applicable regulations and agreements. This kind of process is, however, complex and expensive mainly due to the lack of appropriate guides and tools [97].

Furthermore, privacy breaches are not only originated by external attackers, they can also be originated by internal actors. In fact, an important number of data breaches (around 30%) is originated by trusted and authorized actors [32]. The severity of these threats and the lack of appropriate access and usage control mechanisms is pushing most organizations to strongly limit access and exploitation of data, even internally, making a large part of the information unavailable, and reducing the potential exploitation.

These issues create a strong need for new access control models able to dynamically evaluate these threats and to take access flexible decisions based on the best trade-off

between expected risks and benefits.

The aim of this thesis is to motivate, design, implement and validate a novel privacy-aware risk-based access control model capable to balance the risk and benefits when evaluating access requests. Our model also proposes the application of adaptive adjustment strategies to lower the risk or provide trustworthiness guarantees (i.e. guarantees that the granted access will not be misused). These operations aim to increase the flexibility of the access control process and enhance exploitability of the data while maintaining control over privacy risk.

To identify and evaluate privacy threat scenarios, we also develop and validate a methodology for evaluating privacy violation risk (EPIC). For practical reasons our methodology is designed in the context of cybersecurity, however, it can be adapted to conduct a privacy threat evaluation in other contexts.

The research presented in this thesis was done partially in the context of the SECENTIS Project [1] (The European Industrial Doctorate on Security and Trust of Next Generation Enterprise Information Systems), financed by the European Union grant 317387, under FP7-PEOPLE-2012-ITN and held in collaboration between the Fondazione Bruno Kessler, SAP Security Research at Labs France, and the University of Trento. As in the scope of the industrial doctorate, the research has been strongly based on industrial needs.

1.1 Objectives and research challenges

This thesis aims to achieve two main goals: *i*) Identify and study data breaches and privacy threats by providing tools to systematically conduct a privacy violation threat evaluation process in an enterprise setting. *ii*) Develop and validate solutions to mitigate the identified privacy violation breaches in this environment while preserving data availability and utility. To accomplish these goals, we address the following research questions (***RQ***), with associated challenges (***Cs***).

RQ.1 How can we carry a privacy violation threat evaluation in a meticulous but practical way? Which aspects should be considered, and which parties (e.g., stakeholders, experts) should be involved in the process?

Cs.1 Efficiently evaluating threats scenarios is a complex and time-consuming task [50], due to various factors, mainly: There is a very large number of heterogeneous aspects to be considered (e.g., the system's architecture, data flows, nature of the data, human and organizational factors). This kind of information might not always be available and must be discovered (e.g., some procedures are very specific to a certain type of organizations and they can be different or non existent in others) and investigated (e.g., collected through interviews, extracted from diverse types of

[1] www.secentis.eu

documentation) for the threat evaluation purpose, and therefore several aspects can be overlooked. In addition, such process requires the setting up teams of several experts [120] with different skills and backgrounds, which results in challenging and costly projects.

RQ.2 How to evaluate privacy threats risk? Which metrics to use at the organization level? How can we balance the privacy risk with the best exploitation of data?

Cs.2 Despite the existence of several formal privacy metrics proposed to estimate the likelihood of an adversary of learning a private sensitive information when getting access to a given dataset, none of them in isolation seems appropriate to measure the general privacy violation the risk incurred by an organization when processing sensitive data. The reason is that the validity of these metrics often dependents on specific assumptions on the considered adversary and/or data sharing model (some examples are provided in [15, 56]), while enterprise systems (i.e., cybersecurity system, enterprise resource planning) have many different components that process and store heterogeneous data, complex architecture and data flows, and interactions (including data access) with users with different roles. Therefore none of the existing privacy metrics can be used in isolation to measure the general privacy violation risks involved in running such systems. This complexity calls for a principled but more high-level approach to privacy threat assessment. The prioritization of privacy threats is yet another challenging aspect of threat evaluation. Similarly to security threats, the mitigation of privacy threats often requires costly operations to be implemented [97] and it is crucial for the organization to prioritize its action plan according to the urgency of the threats to handle. Threat prioritization is not a trivial task and it needs to be carefully conducted to avoid misleading misconceptions. Indeed organizations generally tend to underestimate or miss-identify threats (security and privacy threats) involving insider threats [45], which is very dangerous the organization since these threats are real, represent an important percentage of the overall number of threats, and they are quite difficult to identify and handle. For example, the US State of Cybercrime Survey reported in 2014, that around 1/3 of the total number of incidents registers that year were perpetrated by insiders and had more damaging impacts than external attacks [32] (slightly different numbers were reported in 2017 [33]).

RQ.3 How to mitigate the data breaches and privacy violation threats coming from insider authorized actors without hindering their capability to fulfill their business tasks, which need data access?

Cs.3 Unauthorized access to data through external attacks can usually be handled by addressing the security and/or organizational controls and eliminating the leakages

at the origin of the privacy threats [73]. Although this might be a challenging task addressing threats coming from authorized access can be even more challenging. For this kind of actors, we need to apply the data minimization principle when granting access to the data but this is not an easy task since we need to evaluate the access needs and risks both which can depend on a number of factors (e.g., trustworthiness of a requester, security context of the query) that can not be accurately assessed off-line and need to be re-assessed for each access.

RQ.4 How to accurately measure trust and risk in the context of a privacy-aware risk-based access control?

Cs.4 Quantitative risk and trust values are well known to be very hard to compute [16, 111]. Indeed the diversity of risk scenarios, the intangible nature of trust, and the limited amount of historical data for incidents makes an accurate quantitative assessment extremely difficult. In a very restricted context, a fairly accurate qualitative assessment based on domain knowledge and privacy expertise can be however this kind of assessment is not suited for access control where we need a real-time assessment for diverse types of access requests.

RQ.5 How to evaluate the feasibility efficiency of our solution performance, impact on data quality and on privacy improvement of the privacy-aware risk-based access control system?

Cs.5 While evaluating the performance might be a relatively easy task, it is much harder to assess the impact of our solution on the quality of data (or data utility) and evaluate the improvement it brings in terms of privacy and this is mainly due to the lack of appropriate metrics for both aspects. Indeed the utility and privacy is a very hard concept to quantify in a general context (for the reasons explained in ***Cs.2***).

1.2 Contributions

The contributions made by this thesis to address the aforementioned research questions and challenges can be summarized as follows:

1. We provide a methodology for Evaluating Privacy violation rIsk in Cyber-security systems (EPIC). It is a four-steps methodology designed to guide a privacy expert, with the collaboration of security experts from the organization running the system, to the identification of the main privacy threats, and to the assignment of a privacy violation risk value to each of them. EPIC supports both qualitative and quantitative risk values. The resulting evaluation can be used to prioritize mitigation actions to

achieve legal compliance as explained in point a) above. Since the training, and more generally trust, in a specific personnel role, is not considered until mitigation task prioritization, the evaluation is useful for point b) as well. Finally, addressing also point c) above. This contribution is detailed in **Chapter 3** and addresses (***RQ.1***) and (***Cs.1***)

2. We provide (in EPIC) a way to assess the priority of a privacy threat by considering the trustworthiness of the adversary together with the privacy violation risk. This overview (privacy threats details, risk, and priority) helps planning and elaborating the activities necessary to mitigate the risk of the identified privacy threats. This contribution is also detailed in **Chapter 3** and addresses (***RQ.2***) and (***Cs.2***)

3. We develop a novel access control model where access decisions are based on a tread-off between the request's trust with risk. Risk and trust are computed at run-time taking into consideration a diverse number of factors to support access control in dynamic contexts. When the risk is too large compared to the trust level, we propose adaptive adjustment strategies that can decrease the risk and/or increase the trust level to enhance the flexibility of the model while maintaining an acceptable level of protection. The general model is described in **Chapter 4** and addresses (***RQ.3***) and (***Cs.3***).

4. To support diverse data usage scenarios we propose two different privacy-aware implementations of the general risk-based model in the context of privacy. The first model in (**Chapter 5**) is based on syntactic anonymity metrics. The second (in **Chapter 6**) is based on differential privacy. In each model, we provide concrete and understandable risk and trust assessment models to evaluate access, as well as adjustment strategy to enforce the access decision. Both models were implemented and evaluated using relevant case studies. This contribution addresses ***RQ.4***) and (***Cs.4***).

5. We provide in Chapter 5 and Chapter 6 relevant industrial case studies against which we assess the performance and utility of the privacy-aware risk-based access control model. Moreover, in Chapter 7, we will use the privacy threat evaluation methodology, EPIC, to assess our model from the privacy perspective. Indeed EPIC can be used to compare different systems in terms of privacy implications, it can also be used to evaluate the impact of a privacy-preserving solution implemented in a system by comparing the variation of the risk levels of privacy threats identified in a cybersecurity system before and after the adoption of the privacy-preserving solution. These contributions address (***RQ.5***) and (***Cs.5***)

1.3 Structure of the thesis

Figure 1.1 describes the different steps followed during the thesis. It also indicates in which chapters of this dissertation we report each step. In **Chapter 2** we discuss the main related work in the areas of privacy threat assessment and risk- and privacy-based access control. The work reported in this chapter is orthogonal to different steps of Figure 1.1.

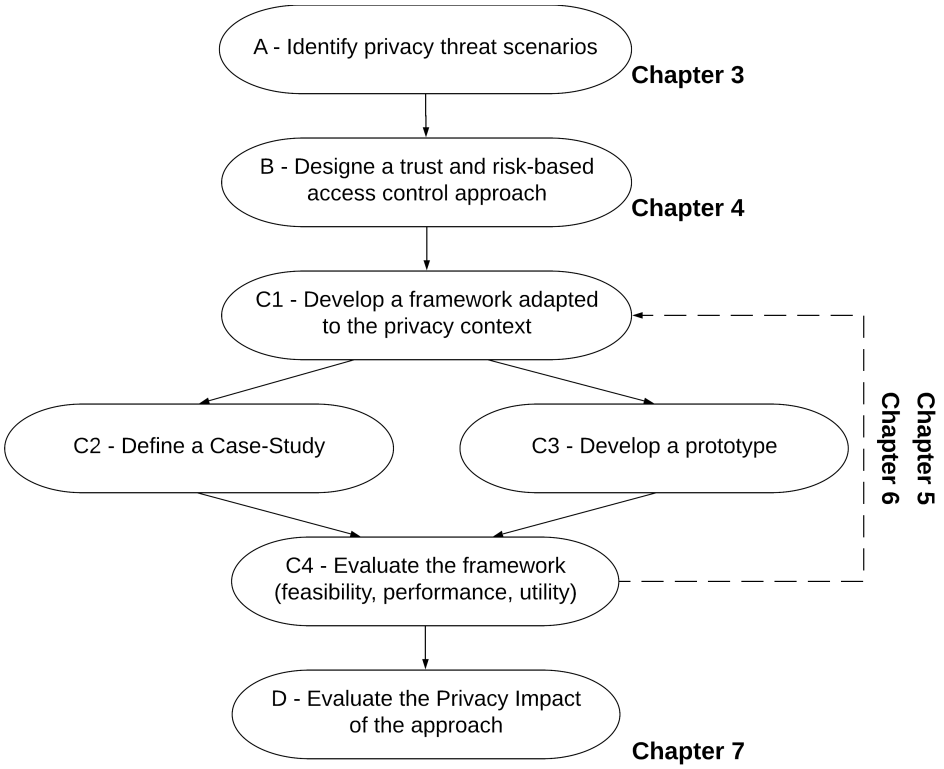


Figure 1.1: Structure of the thesis

In **Chapter 3** we try to identify and study different privacy threat scenarios in the information system of an organization (step A). Since this is a difficult enough task, we chose to focus on the cybersecurity systems a vital component of every organization's information system. To this end, we develop and validate "EPIC" a privacy threat identification and evaluation methodology for cybersecurity systems. In **Chapter 4** we design and evaluate a novel access control model that combines trust with risk and supports access control in dynamic contexts through trust enhancement mechanisms and risk mitigation strategies (step B). We adapt this trust and risk-based access control model to the context of privacy and propose a privacy-aware risk-based access control In **Chapter 5**. This privacy-aware model uses syntactic anonymity metrics to assess the privacy risk. Another category of privacy metrics equally interesting exists in the literature, the differential private met-

rics. In **Chapter 6** we propose a privacy-aware risk-based access control using these metrics. Each the models proposed in these two chapters where developed and evaluated through several iterations of steps *C1*, *C2*, *C3* and *C4*. In **Chapter 7** we evaluate the privacy-aware access control approach impact on privacy by using the EPIC methodology (step *D*). Since this thesis is part of an industrial doctorate program we dedicate the **Chapter 8** to discussing the impact of our work in terms of possible migration to industry, standardization bodies, and open source communities. **Chapter 9** concludes this dissertation and discusses some directions for future work.

1.4 List of publications

1. Chapter 3: [102] Sergio Mascetti, Nadia Metoui, Andrea Lanzi, and Claudio Bettini. Epic: a methodology for evaluating privacy violation risk in cyber security systems. *Submitted to: Transactions on Data Privacy*, 2018.

2. Chapter 5: [110]Nadia Metoui, Michele Bezzi, and Alessandro Armando. *Risk-Based Privacy-Aware Access Control for Threat Detection Systems*, pages 1–30. Springer Berlin Heidelberg, Berlin, Heidelberg, 2017.

3. Chapter 5: [109] Nadia Metoui, Michele Bezzi, and Alessandro Armando. Trust and risk-based access control for privacy preserving threat detection systems. In *International Conference on Future Data and Security Engineering*, pages 285–304. Springer, 2016.

4. Chapter 6: [108] Nadia Metoui and Michele Bezzi. Differential privacy based access control. In *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, pages 962–974. Springer, 2016.

5. Chapter 4: [8] Alessandro Armando, Michele Bezzi, Francesco Di Cerbo, and Nadia Metoui. Balancing trust and risk in access control. In *On the Move to Meaningful Internet Systems: OTM 2015 Conferences*, pages 660–676. Springer International Publishing, 2015.

6. Chapter 4 and 5: [10] Alessandro Armando, Michele Bezzi, Nadia Metoui, and Antonino Sabetta. Risk-based privacy-aware information disclosure. *Int. J. Secur. Softw. Eng.*, 6(2):70–89, April 2015.

7. Chapter 4 and 5: [9] Alessandro Armando, Michele Bezzi, Nadia Metoui, and Antonino Sabetta. Risk-aware information disclosure. In *Data Privacy Management, Autonomous Spontaneous Security, and Security Assurance*, pages 266–276. Springer, 2015.

Chapter 2

Related Work

The research areas most directly related to the work presented in this dissertation are *i*) privacy and security threat assessment (Section 2.1); *ii*)context-aware and risk-based access control systems (Section 2.2); and privacy-preserving access control (Section2.3). In this chapter, we will present an overview of the related work each of these three areas and discuss similarities and differences with our work.

2.1 Privacy threat identification and evaluation

A lot of research has been conducted in the last decades on the identification of privacy threats related to the use of technology, on mitigation techniques, and on methods to evaluate the risk of privacy violations. We can distinguish two main categories of approaches:

(a) Formal approaches addressing specific privacy problems, proposing privacy enhancing methods and metrics to quantitatively evaluate the resulting level of privacy.

(b) Methodological approaches for privacy threat identification and assessment.

Research has been focused on general personal data collected as part of different applications including e-health, geo-location apps, social networks, finance, marketing but we are not aware of any research addressing specifically the evaluation of privacy risks in deploying cybersecurity systems.

In the following we will briefly report on the main efforts regarding both categories mentioned above. However, regarding the formal approaches and the related proposed privacy metrics we will motivate why we decided not to follow this route for evaluating privacy risk violation in cybersecurity systems. Instead, we will illustrate how our proposed methodology relates to the more qualitative privacy threat assessments of the second category, and how it was inspired by work done on security threat assessment.

2.1.1 Privacy metrics

Various privacy metrics have been proposed in the literature to estimate the likelihood of an adversary of learning a private sensitive information when getting access to a given dataset (i.e., obtaining the identity of an individual and associated sensitive information). For example, since anonymity prevents privacy violations, several metrics have been proposed to quantify the level of anonymity of a dataset [40, 92, 99]. Extensions of these metrics have been proposed to evaluate anonymity in different data sharing contexts including location-based service requests [19]. However, their value is somehow limited by the problem of evaluating the adversary's knowledge which can determine which information can actually re-identify individuals. When identification cannot be successfully prevented, various sensitive data obfuscation techniques and related privacy metrics have been proposed. Some metrics measure the distortion or generalization applied to the data, and hence the probability of the adversary to infer the actual sensitive information. Other metrics are based on the notion of *indistinguishability* with differential privacy metrics [55] being an example. A quite comprehensive list of the privacy metrics that have been proposed in the literature can be found in [160]. Finally, there are valuable attempts to provide guidance in the application of privacy enhancing technologies (PET), often related to the above-mentioned metrics [15, 56].

Some of these metrics (and related PETs) may be applied also in the context of cybersecurity systems; For example, some anonymity metrics may be used to evaluate how anonymous is a dataset of security alert logs, and some differential privacy notions may be used to measure the probability of privacy leak in releasing a statistically perturbed Web site access log. However, none of them in isolation seems appropriate to measure the general privacy violation risks involved in running a cybersecurity system. This is partly due to the fact that the validity of these metrics is dependent on specific assumptions on the considered data sharing model while typical cybersecurity systems have many different components that process and store data, complex architecture and data flows, and data access by users with different roles. This complexity calls for a principled but more high-level approach to privacy threat assessment.

2.1.2 Security threat assessment methodologies

Before considering privacy assessment methodologies we briefly report some methodologies adopted for security risk assessment since this is a related and more established field of investigation. Security threat analysis is a common step in the secure software development life-cycle. In the literature, we find several tools and methodologies such as the OCTAVE method [30], ISRAM [82], and the Common Vulnerability Scoring System (CVSS) [107] only to cite a few. Among the most widely used, the STRIDE model was

proposed by Microsoft [69] as a security threat identification process, used to assist engineers to consider security aspects during the development of a software product. This process starts by analyzing the information flow within a system and then modeling system's components using Data Flow Diagrams (DFD); a list of possible security threats is identified for each of the components. STRIDE classifies security threats into six categories (Spoofing, Tempering with data, Repudiation, Information disclosure, Denial of service, and Elevation of privileges). This model-based analysis has inspired the methodology that we are proposing. Indeed, we extend the DFD notation to better model the system components and focus on the privacy threat identification for each component. STRIDE is often used with the threat evaluation model DREAD to assess security risks [145]. DREAD proposes to rate security threats by computing a score based on five criteria (Damage, Reproducibility, Exploitability, Affected users, and Discoverability). This score implicitly expresses the likelihood and severity aspects of a security threat. A similar approach is proposed in our methodology for privacy violation risk assessment.

2.1.3 Privacy threat assessment methodologies

The first approaches to privacy assessment were mostly in the form of checklists with the goal of demonstrating legal compliance [42]. Privacy impact assessment (PIA) methodologies emerged later-on to refine these approaches. Several definitions have been given to PIA (see [73, 75, 130]). David Wright in [161] defines PIA as a methodology for assessing the impacts on privacy of a project, and for taking remediation actions to avoid or minimize negative impacts. Several governmental bodies such as the CNIL (France), NIST (USA), ICO (UK) and the EU Art.29 Working Party have proposed various PIA methodologies [44, 57, 73, 118]. These guidelines although very useful to understand the goals of the assessment, do not guide an organization through the specific steps that should be performed. Among the works that contribute in this direction, Oetzel and Spiekermann present a seven steps methodology to support a complete PIA analysis and systematically match the threats and the appropriate countermeasure [120]. However, their approach only considers the impact of a privacy threat and not the probability of occurrence of the threat, which may lead to an incorrect overall risk estimation. Another aspect that has a relevant impact on the effectiveness of the guidelines is their specialization for a given sector. The methodologies mentioned above are designed for a generic privacy assessment, and consequently, they may not be straightforwardly implemented when addressing the problem in a specific context. Indeed, the development of sector-specific PIAs is mentioned among the priorities in recent EU recommendations [57]. We found very few sector-specific approaches, among which a PIA framework for RFID based applications [46], and a PIA template for smart-grid and smart-metering systems [149].

EPIC is not intended to be a complete PIA methodology, rather it focuses on the systematical analysis of technical aspects of CSS and their implications on privacy. Some non-technical aspects are also considered (intervenability, consent, etc.) but they are only evaluated when they can have an impact on the privacy risk. For example, as we show in the following, lack of compliance with existing regulations influences the impact of a privacy risk and hence should evaluate. Still, EPIC is not intended to provide a systematical analysis of these non-technical aspects.

To the best of our knowledge, the only work in the literature that analyses the problem of privacy violations in cybersecurity systems is a survey paper by Toch et *al.* [152] (co-authored by some of the authors of this paper). The survey proposes a new categorization of cybersecurity systems that help the privacy analysts to identify the personal data that these systems may expose to unauthorized parties. Our work builds on this categorization but takes the proposed analysis to a deeper and more operational level with the main goal of evaluating and comparing the risk of the identified privacy threats. Our methodology considers also aspects like the adversary knowledge, the capability to access the data, amount of data leaked, number of users involved, and other factors that determine the impact of a privacy threat. With respect to the survey that considered also cybersecurity systems for new ecosystems like mobile and IoT systems, we focus on organizational cybersecurity systems and test our proposed methods in a case study involving the systems of a large organization.

Besides PIA, other privacy assessment approaches adopted a requirement engineering perspective to promote the privacy by design principles [48,50,98,119]. Among them, the closest to our proposal, despite not being specific to cybersecurity systems, is probably LINDDUN, a privacy threat analysis framework for software-based systems proposed by Deng et al. [50] and based on the STRIDE model [69]. Privacy threats in LINDDUN are identified through potential misuse scenarios (i.e., scenarios in which an adversary can violate privacy requirements upon accessing the data). Unfortunately, the processes of identification and analysis of misuse scenarios are not specified by the methodology but rely on the expertise of the analysts. LINDDUN does not provide a risk evaluation support either. On the contrary, in our approach, we consider as a threat any data disclosure that can reveal sensitive information about a respondent. Our methodology is specialized for cybersecurity systems and hence the identification of threats is well guided by security and privacy factors (e.g., adversaries' capabilities and knowledge, types of exposed data). We also propose a domain-specific risk assessment model evaluating the likelihood and severity of a threat.

2.2 Risk-based access control

Several approaches have been recently proposed to address the limitations of traditional access control models in terms of lack of flexibility, inability to handle contextual information, evaluation of the trustworthiness of users and in managing access risk.

Among these approaches, we first consider the idea of context awareness. Context-aware access control models (see, e.g. [1, 17, 22]) propose the use of contextual and environmental information (e.g. Spatio-Temporal information) to achieve fine-grained access control. Although these models do not evoke an explicit notion of access risk, the request's context and environment can provide relevant information that could be used to assess the access risk. In our risk-based access control model (presented in Chapter 4) we also consider contextual information when evaluation the access request, but, instead of statically including the contextual condition in the policy we use contextual information as a parameter to compute the trustworthiness of a request, i.e., we indirectly use this information to balance risk.

In addition to context awareness, other works propose to increase the access control flexibility by taking in consideration operational need. McGraw [104] (and later Kandala *et al.* [81]) presents a Risk-Adaptable Access Control (RAdAC) mechanism that determines access decision based on a computation of security risk and operational need. Multiple factors are used to determine the risk and operational need for every request (e.g. user trustworthiness, the sensitivity of the information requested, user role and privileges, level of uncertainty and history of access decisions). This model allows adapting the decision thresholds such that operational needs may outweigh security risk when appropriate, but it does not itself specify any risk model. In our model (see Chapter 4) we also propose dynamic risk thresholds considering business needs among other factors. We include these factors in the trustworthiness of the request following this reasoning: if a requester needs to access a resource to accomplish a business task the likelihood the access is misused is lower and the request is more trustworthy. In addition, our approach provides (according to the organization preferences) the possibility to enhance this trustworthiness level and allow more permissive access when this is required. These trust enhancement strategies require the fulfillment of obligations providing assurances that the access will not be misused (e.g., monitoring of the access) and to mitigate potential misuse impacts (e.g., create a back of a modified data).

More dynamic approaches take both risk and trust in consideration in risk-aware access control (e.g. [34, 35, 37, 52, 144]), In these models each access request or permission activation, the corresponding risk is estimated and if the risk is less than a threshold (often

associated with trust) then access is guaranteed, otherwise, it is denied. Cheng et al. [37], following the multi-level-security paradigm, compute risk and trust thresholds from the sensitivity labels of the resource and clearance level of the users. They also consider what we define a trust enhancement mechanism that provides users with a limited amount of *tokens*, which allow them to access resources with a risk higher than their trust level. The details on how this mechanism can be applied in real cases are not provided. In another work, Chen et al. [34] introduced an abstract model which allows role activation based on a risk evaluation compared to predefined risk thresholds. Trust values are considered, and they have an impact on the risk calculation (decrease the risk). If the risk is too high, the model includes mitigation strategies, indicated as (system) obligations. The paper does not specify how to compute the risk thresholds, trust, and the structure and impact of obligations. In a derived model [35], mitigation strategies have been explicitly defined in terms of user obligations in addition to system obligation. A user obligation describes some actions that have to be fulfilled by the user to get access. Although the model does not consider explicitly trust, it introduces the concept of *diligence score*, which measured the diligence of the user to fulfill the obligations (as in behavioral trust model) and impact the risk estimation.

Following the original Chen et al. [34] model, these papers consider trust as part of the risk value. As a consequence: *i)* trust enhancement and risk mitigation strategies are mixed, and it becomes difficult to find an optimal set of strategies to increase access, keeping risk under control, *ii)* trust thresholds become dependent on the risk scenario, decreasing the flexibility in presence of multiple risk factors. Our model solves these issues by clearly separating trust aspects from risk.

These approaches offer an important improvement in terms of flexibility compared to traditional systems, however, these models still rely on the binary answers "allow" and "deny". Our model proposes a third outcome which is a partial access according to the trust and risk levels. This access can be limited in time (e.g., accessing the resource for one hour) or (in the case of data) granted to partial or anonymity views of the requested data. This can provide limited but useful access with a lower risk. In addition, the above approaches do not provide a concrete way to assess the risk and trust, nor concrete risk mitigation strategies. In this thesis in Chapter 5 and Chapter 6 we provide a concrete risk assessment models, in the context of data privacy, leveraging well-established privacy metrics such as k-anonymity and differential privacy. We also propose to use anonymization techniques to enforce the risk mitigation prior to granting access.

2.3 Privacy-preserving access control

In Chapter5 and Chapter6 we propose two privacy-aware access control models aiming to preserve privacy when querying a sensitive dataset, Indeed, privacy issues are lately receiving growing attention and several access-control-based privacy-preserving approaches have been proposed in the literature. These works can be classified into two categories:

(a) Approches to preserving the privacy of requesters: These approaches aime to protect (hide) the identity of data requesters [7, 78] and/or ensure the confidentiality of their request [29, 165, 166] when access control is evaluated and the responce is sent back

(b) Approaches to preserving the privacy of data-owner: These approaches aim to protect the privacy of respondents/data-owner when granting access to a sensitive data by applying one or several of the following principles "*anonymity*", "*pseudonymity*", "*unlinkability*", "*linkability*", "*undetectability*", "*unobservability*"

The work presented in this thesis is more related to approaches in category b). In this category we identified two interesting types of privacy-preserving access control "policy-based model" and "risk-based models"

Policy-based privacy-preserving access control In this category, some works propose to extend existing access control models by adding conditions and obligation to enforce privacy paradises such as access purpose, limitation of use, quality of data etc. For instance Martino *et al.* introduce in [101] a family of models (P-RBAC) Privacy-aware Role Based Access Control models that extend the RBAC model by adding privacy-sensitive data permission granted according to the purpose of access and in return of obligations to be fulfilled. Byun et Li proposed an access framework for privacy-preserving access control systems [27, 28] based on the notion of purpose. Intended purposes are associated with data in order to regulate data accesses and play in a certain way the role of privacy policies. Access purposes are the requester purposes to access an data item so when an access to a data item is requested, the access purpose is checked against the intended purposes for the data item. These approaches offer a good support for expressing privacy-related organizational policies and allow the enforcement of these policies within an access control module. However, they don't offer privacy concrete privacy guarantees (e.g., guarantees that the data will not be misused, re-used for other purposes after release) nor they can guarantee the enforcement of the privacy-preserving obligations. In addition similarly to traditional access control, these approaches increase the rigidity of the access control system by adding (often non-negotiable) privacy constraints which limited furthermore the availability of the data.

Unlike these approaches, we use a formal guarantee for privacy using syntactic anonymity metrics(see [40,92,99] for review) in the model presented in Chapter 5; where access risk is computer and mitigated to meet the required privacy level. And using differential privacy (see [43,54] for review) in the model presented in Chapter 6. In the latter, we do not explicitly compute the risk, since it is very hard to estimate risk for a data set generated by a differentially private mechanism [103]. However, after mitigation our access control system guarantees the release of (ϵ) data [90].

Other works propose to consider users privacy preferences when controlling access to their data or when this data is used. This would be achieved through privacy policies and adequate architecture to enforce them, along with access control, in several scenarios (e.g., a third party handling the data, secondary applications re-purpose or re-use the data). Ardagna *et al.* (in [4–6]) present a privacy-aware framework that integrates access control policies together with privacy policies regulating how personal identifiable information should be handled by the requester. These policies are established during a negotiation between different parties and partially enforced by each of them, which requires apriori knowledge of the requester or a complex dialog between parties. A similar approach is based on the concept of sticky policies [125], in this framework, privacy policies, expressing users preferences for data handling, are attached to the data, enabling to improve control over the usage of personal information and to define usage constraints and obligations as data travels across multiple parties (e.g., in the cloud). These policies are enforced at data consumer's level.

These approaches are very interesting in the context where the data-owners have some control over thier which is often the case when the data is directly collected (the data-owner provides the data to the data handler). However they are not adequate when data is indirectly collected and the user has no control over it (or sometimes has no knowledge this data is collected), which is often the case in several organizations IT systems e.g., client management and human resource systems (example described in Chapter 6), cybersecurity systems (example described in Chapter 5). Our approach instead is not based on data owner's preferences but establishes and enforces quantitative privacy thresholds to guarantee the privacy of respondents. We developed these thresholds considering the urgency of the context (need-to-know) and some considerations legal requirements on privacy and labor regulation (in the context of employees data).

Besides both approaches presented above assume the requester is trusted (up to a certain degree) since they partially delegate the privacy policy enforcement to them. In our model, we actually assess the trustworthiness of the requester as part of the access control assessment. If a need for a more permissive access can be justified we propose trust enhancement techniques to allow this access in return of fulfillment of certain obligations. These approaches also present the traditional access control flexibility issue, offering only a binary all-or-nothing response since they do not consider anonymization, nor other risk

mitigation strategies. In our model, we propose to use anonymization (among other techniques) as risk mitigation strategy with the goal to increase the flexibility

Risk-based privacy preserving access control To the best of our knowledge, risk-based approaches to privacy-preserving access control have been barely explored in the literature. In [156, 157] Ulltveit-Moe et *al.* propose to assess the likelihood of privacy violations in intrusion detection systems (IDS) based on information entropy in network information flow. Then, they use this measure to differentiate between rules (IDS rules) with a high likelihood of privacy violation and rules with low ones. They also propose to modify rules with high privacy violation likelihoods or restrict access to sensitive data (on strict need-to-know approach) and use anonymization to implement these restrictions. When this information is accessed by security agents (human agents) to monitor the IDS alerts. This approach proposes to set two profiles of users according to the expertise level: the first profile allows monitoring tasks using anonymized data the second consists of security experts, with clearance to perform necessary privacy-sensitive operations to investigate attacks. However Ulltveit-Moe et *al.* do not elaborate how this access control is implemented, or how it behaves according to the likelihood of violation. The entropy-based privacy leakage metric they propose is very interesting, however, the violation likelihood/risk needs to be computed off-line for each rule (prior to the access control) based on already existing information in the IDS alarm database which might lead to assessment mistakes depending on the database. Moreover, this model clearly increases the privacy protection but it might be difficult to apply in realistic cases in the context of cybersecurity because the risk mitigation relies on anonymizing the entire (source) dataset beforehand, resulting in either low privacy or low utility.

Indeed privacy is a big issue in these cybersecurity systems (and generally in cybersecurity) since network log and security log data used to monitor the information system and detect security threats often contain very sensitive data. In Chapter 5 in Section 5.5 we also tackle privacy violation scenarios threat detection systems (TDS). In our approach, similarly to Ulltveit-Moe et *al*'s approach, we aim to optimize the application of the need-to-know principle. However, unlike them, we offer adaptive adjustment strategies, that according to the priorities of the context, allows to mitigate the risk or provide trustworthiness guarantees that the granted access will not be misused. These operations aim to increase the flexibility of the access control process and enhance exploitability of the data while maintaining control over privacy risk Besides our model that can be integrated to well established access control models such as RBAC (example in Section 5.3) as ABAC (example in Section 5.5)

In the same context specific anonymization techniques for logs where proposed in [112]. We implemented several of the proposed anonymization techniques in a prototype of our privacy-aware access control model (described in Section5.6), and, although based on k-anonymity, our framework can include other privacy measures by changing the risk function. More specifically, entropy-based privacy metrics can be easily integrated with k-anonymity approach, as shown in [87].

Chapter 3

Threat scenario and mitigation strategies

In this chapter, we would like to investigate (identify, evaluate and prioritize) the privacy threats in data-driven systems. This analysis aims to understand the implication of insider authorized actors in potential privacy threats. It will also help assess the impact of these threats and select the most appropriate mitigation strategies. As an example of data-driven systems, we chose to focus on Cyber Security Systems (CSSs).

CSSs play a fundamental role in guaranteeing data confidentiality, integrity, and availability. Modern CSSs relay more and more on big amounts of data collected by sensors (e.g., agent installed in end-user machines) deployed all over the network or the information system to protect. The data are, then, sent to central nodes (e.g., IDS Intrusion Detection Systems, SIEM Security Information and Event Management) to undergo different kinds of analysis. This centralized way of monitoring allows for having a wide perspective of what is happening on the information system than other (isolated) security products. Therefore, it enables better identification and faster reaction to increasingly complex cyber-security threats. However, while processing the data, CSSs can intentionally or unintentionally expose personal information to people that can misuse them. For this reason, privacy implications of a CSS should be carefully evaluated. This is a challenging task mainly because modern CSSs have complex architectures and components. Moreover, data processed by CSSs can be exposed to different actors, both internal and external to the organization.

Consequently, we needed to develop a new methodology, specifically designed the evaluate privacy violation risk in cyber-security systems. Differently, from other general purpose guidelines, our methodology (called EPIC*) is an operational*

methodology aimed at guiding security and privacy experts with step-by-step instructions from modeling data exposure in the CSS to the systematical identification of privacy threats and evaluation of their associated privacy violation risk.

3.1 Introduction

Privacy policy makers and data protection authorities all over the world are considering the impact on privacy of the large amount of identifiable sensitive data that are being collected and processed by public and private organizations. This is mainly the result of the adoption of new technologies like mobile and pervasive systems, social networks, and big data analytics, but also the evolution of technologies applied in surveillance and cybersecurity systems. An example of regulation activity motivated by these concerns is the EU General Data Protection Regulation, adopted in May 2016 [150]. While regulations differ in different countries, some general principles are shared; for example, user informed consent remains a pillar, and de-identification, despite the limits of anonymization techniques, is still considered a mean to avoid or, at least, mitigate privacy violation risk [66]. Another shared recommendation to organizations deploying complex automated processes handling large amounts of personal data is to systematically and thoroughly analyze how the process affects the privacy of the individuals involved and evaluate the risks in order to identify appropriate mitigation actions. This analysis is often called Privacy Impact Assessment (PIA) and it is, in some cases, a legal obligation as a necessary element in a *privacy by design* approach. However, its value goes beyond the design phase since it is also highly valuable when evaluating the compliance of already existing systems as well as when comparing the privacy risks of alternative systems.

Several documents exist guiding the experts in privacy impact assessments, but they usually consist of high-level guidelines instead of step-by-step instructions, partly motivated by the fact that they are sector independent. Therefore, the importance of designing sectoral PIA methodologies emerges in recent documents by EU data protection authorities [57]. In most cases, the interest is currently focused on sectors like healthcare, e-commerce, finance, and insurance, and less attention is paid to cybersecurity systems. These systems handle large amounts of sensitive information as, for example, the data obtained by monitoring employees personal computers, mobile phones, and the whole organization network traffic [57]. In the last decade, cybersecurity systems have been increasing their strategic role for the protection of the IT infrastructure of industries and organizations. The wide adoption of digital technologies to control even critical infrastructure and the extension of organizational IT systems to include mobile and IoT devices have increased the attack surface and the impact that cyber attacks can have. This led to

a significant increase in the complexity of cybersecurity systems in terms of components, architecture, amount of data being analyzed, and personnel involved in managing the systems.

The role of CSS with respect to privacy is twofold. On one side, CSS are an essential tool to prevent privacy violation, e.g., by avoiding unauthorized access to data. On the other hand, CSS often process a large amount of personal data, e.g., by monitoring network traffic, and hence they can pose a privacy threat. For example, consider a security administrator that discovers the sexual orientation of a colleague while reading email logs during the investigation for a security incident. In general, privacy leaks from CSS can lead to discrimination in the workplace affecting both the relationships among colleagues and between the employee and the management, including effects on professional carrier. Privacy leaks from CSS can also affect external subjects, e.g., customers, with effects similar to the ones resulting from the release of private data through different channels. This includes unsolicited advertising, and even more serious problems like identity theft, blackmailing, and physical assaults. These consequences have an indirect impact on the organization running the CSS which is responsible for properly handling private data.

An accurate evaluation of privacy violation risks in a cybersecurity system is important for at least three reasons:
a) it identifies the gaps with respect to the applicable regulation, so that appropriate remediation actions can be taken to achieve compliance;
b) it shows the responsibility of personnel like security, system, and network administrators in terms of personal data access, suggesting role-specific training and screening;
c) it highlights data collection practices that may make employees worry about their privacy and as a result, it can be an incentive for them to circumvent some of the cybersecurity mechanisms.

In the following, we present the *EPIC* (Evaluating Privacy violation rIsk in Cybersecurity systems) methodology. EPIC is composed of four steps designed to guide a privacy expert, with the collaboration of security experts from the organization running the system, to the identification of the main privacy threats, and to the assignment of a privacy violation risk value to each of them. The proposed methodology supports both qualitative and quantitative risk values, the latter being preferable when it is possible to quantitatively assess how much a privacy threat would impact on the organization, for example in terms of monetary loss. The resulting evaluation can be used to prioritize mitigation actions to achieve legal compliance as explained in point a) above. Since training, and more generally trust, in a specific personnel role, is not considered until mitigation task prioritization, the evaluation is useful for point b) as well. Finally, our methodology can be used to compare different cybersecurity systems in terms of privacy implications, and possibly to design new cybersecurity systems that can effectively combine built-in

privacy preserving features with protection from cyber attacks, addressing also point c) above. The methodology is illustrated through a running example and then applied in a use case considering the actual cybersecurity system of a large academic organization managing over 15,000 hosts. In the last part of this chapter, we will analyze the results obtained from the use case and discuss adequate mitigation strategies for different categories of threats.

This Chapter is structured as follows. We describe our privacy violation risk evaluation methodology and explain its three first steps n Section 3.2. Section 3.3 is dedicated to the fourth step of the methodology dealing with the assignment of risk values and prioritizing mitigation actions, and Section 3.4 to the application of the methodology to the selected use case. in Section 3.5 we will discuss possible threat mitigation strategies for each different categories of adversaries involved in the identified threats. We will conclude with a discussion in Section 3.6.

3.2 Methodology

3.2.1 Overview

The EPIC methodology is organized into four steps as illustrated in Figure 3.1. The whole process requires the participation of a team, involving members with different expertise, namely privacy, and security, as well as personnel of the organization in which the CSS is deployed.

Security experts of the team have a major role in Step 2 while privacy experts take the lead in Step 3 and Step 4. Step 1 (modeling the CSS) and Step 2 (identifying data exposures) require the collaboration of personnel of the organization in which the CSS is deployed. Indeed, information about the actual configuration of the CSS, the processes involved, as well as about the structure of the organization including users, system, network and security personnel must be acquired. In the following we use the term **expert** to refer to a person that contribute to the analysis following the methodology.

The results of obtained at the fourth step of this methodology can be used to compare the privacy level in different cybersecurity products. It can also be used to initiate a privacy threat mitigation phase where privacy enhancing solutions are selected, or developed, and implemented to lower the privacy violation risk of identified threats

3.2.2 EPIC First Step: Model the cybersecurity system

The first step of the methodology aims to model the specific CSS under investigation. This step is particularly relevant for two reasons. First, we can expect that some of the experts involved in the privacy threat modeling process do not have the required

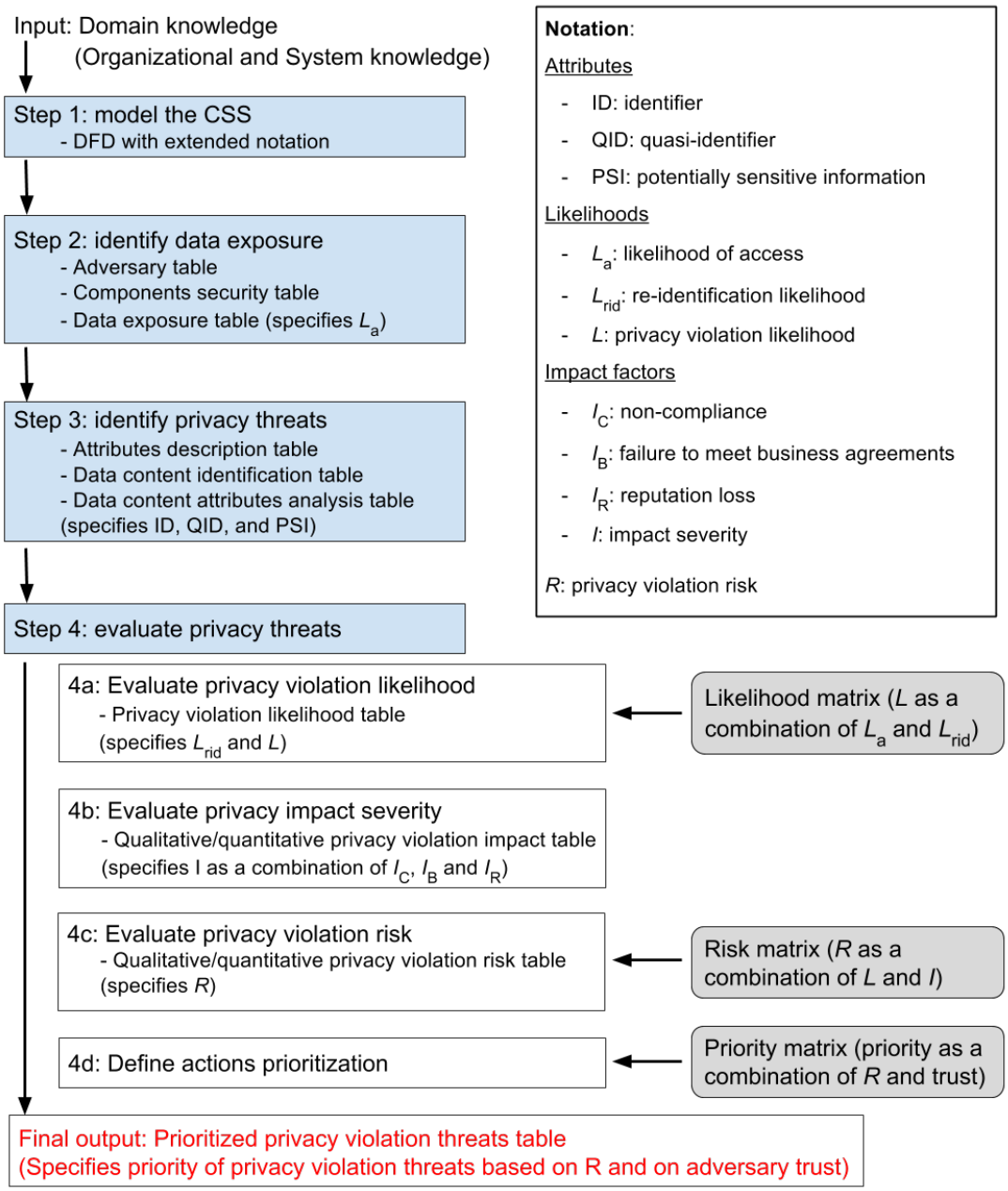


Figure 3.1: Methodology organization in four steps.

knowledge about the system. For example, privacy experts are not expected to know which are the components of the CSS, how data flow in the system and which actors are involved. Second, an explicit system description helps the experts to collaborate and prevents misunderstandings among them. In our use case, this step was completed by members of our team supported by system and security administrators from the institution running the CSS. Modeling a CSS as part of Step 1 must include the following aspects.

- System aspects: overall architecture and control processes.
- Data aspects: data flow, data type, and data storage.
- Functional aspects: users, roles, and functional processes.

A well-known formalism to represent data and functional aspects is Data Flow Diagram (DFD) [36]. This formalism allows us to represent five types of elements (see Figure 3.2): data flow is denoted with a full arrow, entities are denoted with a rectangle, storage with parallel line segments and functional processes (i.e., processes implementing the main system functionalities) with a circle. Finally, a double circle is used to represent a complex process i.e., a single component that represents several functional processes.

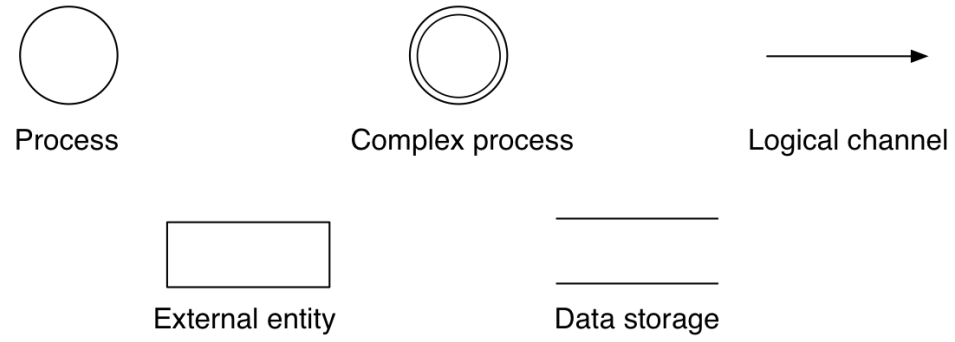


Figure 3.2: Elements of the traditional data flow diagram (DFD)

In this contribution, we extend DFD (and we call it **DFD+**) to also account for system aspects and hence to better detect situations in which data is exposed to an actor. We introduce four additional graphical symbols (see examples in Figure 3.3); a box represents a hardware component, an arrow with a small circle represents a physical channel connecting hardware components, a dashed arrow represents control flow and a dashed circle represents a control process that implements IT controls such as maintenance and security.

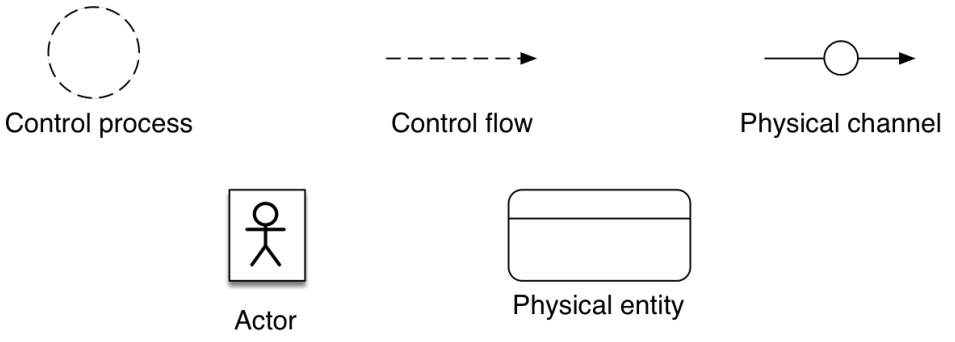


Figure 3.3: Elements of the extended data flow diagram (DFD+)

In Example 1 we illustrate DFD+ and its use in CSS modeling as required by Step 1.

example 1. *Figure 3.4 describes an application level firewall. Data flows from the source entity Network to the destination entity Security administrator. Channel* C1 *shows how data flows from Network to the Firewall hardware component.* C1 *is marked as a physical channel and it is associated with a label (*Network Traffic*) that represents the type of data; in this case, it is the portion of network traffic that should be checked by the CSS. The logical destination of* C1 *is the Traffic Filtering process. Upon detecting a security threat, this process sends the threat description to the data storage* DS1*. Note the different representation of* C2 *with respect to* C1 *due to the fact that* C2 *is a logical channel.*

From DS1 *data flows through the physical channel* C3 *to another hardware component,* Remote Console*, where threat reports are organized for visualization by process* P2*. Then,* P2 *sends this information through physical channel* C4 *to the* security administrator *who is the destination entity and the main actor interacting with the CSS.*

In this diagram we also model a secondary actor system administrator *interacting with the hardware machine hosting the CSS (*Firewall*). The aim of the interaction is* Administration and Maintenance *and indeed* CP1 *is marked with a dashed circle representing a control process. Similarly, the dashed arrows represent a control flow. Another control process (*CP2*) allows the* security administrator *to manage data storage* DS1*.*

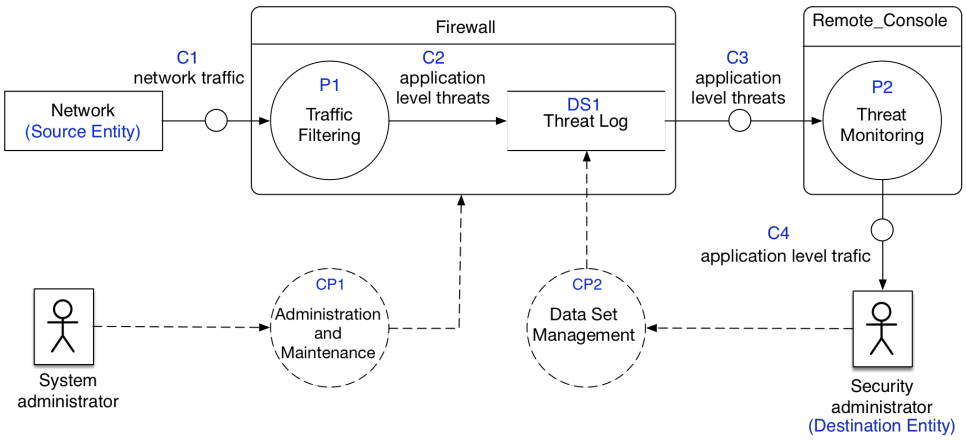


Figure 3.4: CSS modeling with DFD+ (running example).

3.2.3 EPIC Second Step: Identify data exposure

The aim of the second step is to systematically identify all possible data exposures, i.e., situations in which data is disclosed to a potential adversary. A **data exposure** (or **exposure** for short) is identified by the component that is leaking data and by the adversary that can access that data; it is also characterized by other attributes that we specify in this section. **Component** refers to channels, processes and data storages identified in

Step 1. The term **adversary** refers to an actor identified in Step 1 as a subject normally interacting with the CSS or other people, which can either be external adversaries (e.g., a hacker violating a machine and accessing a data storage) or internal ones (e.g., a network administrator or other employees). An **adversaries table** (Table 3.1) containing a list of adversaries, each associated with a brief description, needs to be identified at the beginning of this step. In Table 3.1 we report this list considering our running example.

Table 3.1: Adversaries (running example)

Adversary	**Description**
Security administrator	Their main tasks are to perform monitoring and investigation as well as the maintenance and configuration of the data storage (e.g. add, modify roles and privileges).
System administrator	Their tasks include maintenance of the system hosting the *Firewall* (e.g., troubleshooting, installing updates software/firmware)
Network administrator	Their main task is to ensure the correct functioning of the network (routing, DNS, etc.).
Other internal adversaries	Individuals attempting a nonauthorized access from inside the organization network.
External adversaries	Individuals attempting a non authorized access from outside the organization network.

While the organization management and owner, in principle, may also be an adversary, they usually do not have direct access to the system and the risk of them violating privacy can be easily evaluated by combining the risks computed for the operators that have direct access, since they are the ones that can take order from them. Moreover, the risk assessment is performed on their behalf and in their interest. This is similar to IT security threat modeling: system owners are usually not considered as potential attackers of their own system.

Step 2 also requires, for each component specified in the model, to identify the set of adversaries that can acquire data from that component. More specifically the aim is to identify the adversaries that:

- can access data transiting along a channel (either logical or physical) ;
- can read data from a data storage;
- can obtain data from a process, for example by observing the process output or altering the process behavior.

Clearly, different adversaries need different efforts to obtain data from a component. For example in the DFD depicted in Figure 3.4, the *Security Administrator* has the credentials to access data storage *DS1*, hence the effort is negligible. Vice versa, an external adversary needs to violate a number of security systems and resources are required

to accomplish this task (economical, computational, knowledge). In principle, external adversaries may also obtain data from internal adversaries, and more generally adversaries may collude with each other. However, as in security threat analysis, we first assume that adversaries do not collude. The likelihood that an internal actor shares data with external adversaries is related to the organization policies, legal agreements, and in general to the level of trust in that actor.

We model the difference in the effort required to obtain data from a component through the **likelihood of access** (L_a) parameter, that, intuitively, is inversely proportional to the effort required to access to the component. The likelihood of access only takes into account the technical difficulties that a given adversary has to face to access a component; it does not depend on the willingness of the adversary to maliciously access that component or, in other words, the *trust* we have on the specific person or in personnel acting under a specific role (e.g., network administrators). These aspects are considered in Step 4.

We use the following five values for the likelihood of access:

- *Negligible*: it is technically very difficult for the adversary to access the component and it is highly unlikely that access can be obtained with a reasonable effort;
- *Low*: it is technically difficult for the adversary to access the component and a significant effort is required;
- *Medium*: it is technically possible for the adversary to access the component, but this requires moderate effort;
- *High*: it is technically easy for the adversary to access the component with a limited effort;
- *Authorized*: the adversary is authorized to access the component, hence no effort is required.

The likelihood of access depends on the security mechanisms (e.g., access control, encryption) implemented to protect that component. For this reason, for each component, we list the security mechanisms, together with their details, including, for example, which users are authorized to access through an access policy (enforced by access control system). This is called the **components security table** (see for example Table 3.2).

It is also clear that different exposures have different magnitudes and results in leaking different amount of data. To estimate the **exposure magnitude** different approaches should be used, depending on the type of component.

- *Exposure magnitude in data storage.* The amount of information incoming in the data storage, as well as the retention period of this information, can help estimating the exposure magnitude. For example, if we know that approximately $1,000$ logs

Table 3.2: Components security (running example)

Component	Authorized users	Security Certified	Security mechanisms
DS1	Security administrator	YES	Encryption, access control, authentication, firewall, NIDS
C3	None	YES	Firewall, NIDS, private network
P2	Security administrator	NO	Access control, authentication, firewall, NIDS

are recorded in a data storage daily and that retention period is 30 days, we can conclude that the data storage contains about $30,000$ logs.

- *Exposure magnitude in channels.* When data is exposed through a channel, we should take into account the data throughput (how much data is transmitted in the unit of time) along the channel and an estimation of how long the adversary can listen to the channel.
- *Exposure magnitude in processes.* Similarly to channels, we should take into account how much data the adversary can access. This may depend on how long the adversary can access the process.

The results of Step 2 are reported in the **data exposures table** (for example Table 3.3) that lists, for each combination of components and adversaries, the likelihood of accessing data from that component by that adversary together with the exposure magnitude. Example 2 illustrates an instance of this process and the result is shown in Table 3.3 where a brief motivation is also reported for each row. These notes are very important to communicate with collaborators on the analysis (e.g., security expert) and they are also useful if the analysis has to be repeated again in the future. The motivation field is used in most of the other tables we present in our methodology, especially when the assessment relies on the expert's subjective judgment.

Note that the two leftmost columns of Table 3.3 are derived from previous tables (i.e., adversaries table and components security table) while the four columns on the right include new content. Henceforth we use the following notation: a double line (like between "Adversary" and "Exp." in Table 3.3) distinguishes the previous content (on the left) from the new one (on the right).

At the end of Step 2 all exposures with a *negligible* likelihood of access are cleared (e.g., those highlighted in Table 3.3), while the remaining ones are further investigated in Step 3.

example 2. *This example continues from Example 1 and presents the components security and data exposures tables for three components:* DS1*,* C3 *and* P2*.*

From the CSS model, we know that the security administrator *can access* DS1 *and we report this information in the components security table (Table 3.2). In this example,*

it is relevant to know that the security of DS1 *has been certified, which means that a specific auditing, possibly including penetration attacks, has been performed. We report this information in the table. Finally, we list the security mechanism adopted to protect* DS1*: encryption, access control, authentication, firewall and NIDS. No user is authorized to access channel* C3*, whose security has been certified and that is protected by a firewall, an NIDS, and a private network. Finally, the* security administrator *can access* P2*, whose security has not been certified. This component is protected by access control, authentication, firewall and NIDS.*

Based on the results of the components security table, we now show how to create the data exposures table considering four adversaries: security administrator*,* system administrator*,* network administrator *and* external adversary*. The result is reported in Table 3.3.*

Since the security administrator *has access to the data storage* DS1*, the likelihood of access is reported as* authorized*. Instead, the* system administrator *is not authorized to access DS1 but has access to the physical machine hosting this component. By cracking data encryption (note in the components security table that* DS1 *does implement encryption), the* system administrator *can obtain data from* DS1*, hence we associated this a* medium *likelihood of access. The effort required by the* external adversary *is even higher, as he needs to elude the security protections of the network (firewall, NIDS) to gain access to the machine hosting* DS1*, then bypass the authorization and access control mechanisms and decrypt the data. These security mechanisms have been certified (as reported in Table 3.2) and hence the likelihood of access by the* external adversary *is marked as* negligible*. The likelihood of access by a* network administrator *is also* negligible*. Indeed, since* DS1 *is well configured and security tested, this adversary has to elude all the security mechanisms and make a considerable effort in order to gain access to data from* DS1*.*

Regarding C3*, no user is authorized to access. Since the component's security is certified, we can assign* negligible *likelihood of access to* external adversary*. In this case the* security administrator *needs basically the same effort as an external adversary to access* C3*, so it is also marked as* negligible*. The same does not hold for the* system administrator*, who administers the firewall machine and hence can listen to channel* C3 *with* high *likelihood of access. The* network administrator *has access to the network equipment and can attempt to listen to channel* C3*, thus the likelihood of access is considered* high*.*

Considering the list of security mechanisms protecting process P3*, an unauthorized access attempt from either* system administrator*,* network administrator*, or* external adversary *is very unlikely; however, since these mechanisms were not certified we assign* low *(instead of negligible) likelihood of access to these adversaries for* P3*. The likelihood of access for the* security administrator *is* authorized *as he is allowed to observe the output of* P3 *as part of his security monitoring tasks.*

Table 3.3: Data exposures (running example)

Cp.	Adversary	Exp.	L_a	Exp. Magn.	Motivation
DS1: Threat Log	Security admin.	*Exp*1	Authorized	Important $\approx 100k$ rec	Administrator of the DS (see DFD)
	System admin.	*Exp*2	Medium	*Same as above*	Can Access Mchine But data is Encrypted
	Network admin.	*Exp*3	Negligible	*Same as above*	A Network admin. has to elude the network protection bypass authentication and AC mechanisms and the data is encrypted
	Ext. adversary	*Exp*4	Negligible	*Same as above*	The adversary has to elude the network protection bypass authentication and AC mechanisms and the data is encrypted
C3: application level threats	Security admin.	*Exp*5	Negligible	Limited $\approx 20k$ rec	Need to bypass network protection
	System admin.	*Exp*6	High	*Same as above*	Can Compromise the machine hosting the Firewall and listen to channel *C3*
	Network admin.	*Exp*7	High	*Same as above*	Have access to the Network equipment and can listen to channel *C3*
	Ext. adversary	*Exp*8	Negligible	Very limited $\leq 5k$ rec	The adversary has to elude the network protection, bypass authentication and AC mechanisms
P2: Threat Monitoring	Security admin.	*Exp*9	Authorized	Limited $\approx 30k$ rec	Can observe the output of process P2
	System admin.	*Exp*10	Low	*Same as above*	Should not be able to access, but security has not been tested
	Network admin.	*Exp*11	Low	*Same as above*	*Same as above*
	Ext. adversary	*Exp*12	Low	*Same as above*	*Same as above*

3.2.4 EPIC Third Step: Identify privacy threats

The objective of Step 3 is to determine whether data leaked in each exposure identified in Step 2 can potentially lead to a privacy violation. In order to assess this, we need to take into account what type of data is actually exposed. A given component can expose heterogeneous data. For instance, *DS1* in Example 2 exposes some log records that only contain the IP address of a user as well as others that also include the file being transmitted by that user. Another example is reported in Figure 3.5, showing the user interface of an application-level firewall (PAN-OS 6.1). The upper part of the figure shows results from security threat detection based on URLs filtering while the lower part

Table 3.4: Attributes description (running example)

Name	Description	Domain	Example
IP(out-dst)	The destination IP address of outgoing traffic	IP addresses	216.58.205.195
IP(in-src)	The source IP address of incoming traffic	IP addresses	192.30.253.112
IP(in-dst)	The destination IP address of incoming traffic	IP addresses	132.133.56.45
File	A file being transmitted	String of bytes	

report results from threat detection based on file filtering and the two tables have different attributes.

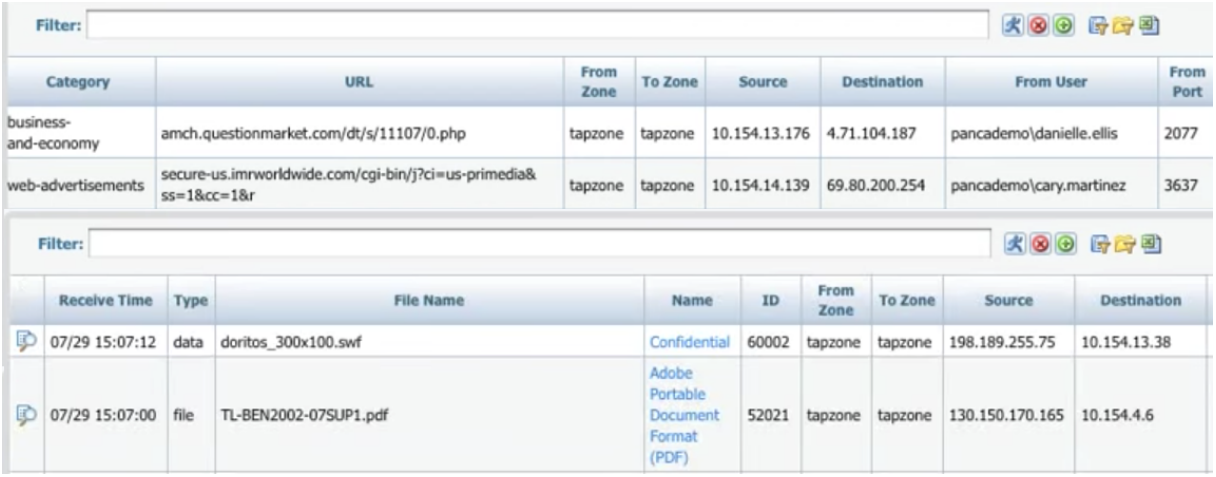


Figure 3.5: PAN-OS 6.1 interface to the logs (from Palo Alto Networks live community video tutorials)

We refer to each data type being exposed as a **data content**, each composed by a set of attributes. The **attributes description table** (for example Table 3.4). lists all attributes exposed in each data content and reports their name, description, domain and some example values. Table 3.4 shows the attributes description table for our running example.

We then associate each exposure (i.e., component and adversary) with the data contents it exposes. This is reported in the **data content identification table** (for example Table 3.5). that presents, for each pair of component and adversary derived from the data exposure table, the likelihood of access (as previously evaluated) and the list of data contents exposed by that component to that adversary. Table 3.5 shows an example reporting some selected exposures from Table 3.3. Note that in Table 3.5 each data content is exposed by each considered component to each considered adversary. This is not always the case as it can happen that two components expose different data contents and that a component exposes different data contents to different adversaries.

We then evaluate whether a combination of exposure and data content represents a privacy threat by analyzing how the adversary can discover the association between a

Table 3.5: Data content identification (running example)

Exposure				Data content
Exposure	**Component**	**Adversary**	L_a	
*Exp*1	DS1. threat log	Security administrator	Authorized	*dc*1: IP(out-dst)
				*dc*2: IP(in-src), IP(in-dst)
				*dc*3 : IP(in-src), IP(in-dst), File
*Exp*7	C3. application level threats	Network administrator	High	*dc*1: IP(out-dst)
				*dc*2: IP(in-src), IP(in-dst)
				*dc*3 : IP(in-src), IP(in-dst), File
*Exp*12	P2. threat monitoring	Ext. adversary	Low	*dc*1: IP(out-dst)
				*dc*2: IP(in-src), IP(in-dst)
				*dc*3: IP(in-src), IP(in-dst), File

sensitive information and an identified respondent. This is clearly related to the semantics of the data being exposed and on the knowledge accessible to the adversary. We first classify the attributes according to the following definitions.

- **Potentially Sensitive Information (PSI):** attribute or set of attributes that can be considered as sensitive. I.e., the combined values of the attributes in each of these sets reveal sensitive information about the data respondent.
- **Identifier (ID):** attribute or set of attributes that uniquely identifies a respondent in a data-set.
- **Quasi-Identifier (QID):** attribute or set of attributes that, combined with other information (including adversary's background knowledge), can be used to identify the respondent in a data-set (or to restrict the set of candidate respondents).

The recognition of QIDs and the related assumptions about background knowledge, also required by most anonymization techniques, is one of the most difficult tasks in privacy protection [20]; however, it becomes more feasible when considering a restricted domain with specific types of data content and adversaries, like the one we are considering. Table 3.6 shows an example of the **data content attributes analysis table** that reports the attributes classification for each data content and also describes the expected adversary's background knowledge. The privacy expert is also expected to motivate or comment the classification of each attribute. These motivations should be reported in the tables delivered at each step. However for sake of brevity, in this chapter, we will not report the motivations in the tables but we report them in the text.

example 3. *In Table 3.6, the attribute* IP(out-dst)*, contained in data content dc*1*, is classified as a PSI attribute . In fact* IP(out-dst) *is the destination IP address of outgoing traffic/request (see Table 3.4); this address can reveal sensitive information about the*

Table 3.6: Data content attributes analysis (running example)

Data content	**ID**	**QID**		**PSI**
		Attribute	**Bg. Kowledge**	
*dc*1: IP(out-dst)	None	None	None	IP(out-dst)
*dc*2: IP(in-src), IP(in-dst)	None	IP(in-dst)	List associating IP-addresses with user-names	IP(in-src)
*dc*3 : IP(in-src), IP(in-dst), file	None	IP(in-dst), file	List associating IP-addresses with user-names	IP(in-src), file

*respondent who sent the request e.g., in case of HTTP traffic this attribute will reveal the domain name of the web page visited by the respondent. dc*1 *contains neither ID attributes nor QID attributes because* IP(out-dst) *does not provide any information about the data respondent in the organization that initiated the communication.*

*Data content dc*2 *contains no ID attributes and a QID attribute* IP(in-dst) *that refers to the destination IP address of incoming traffic (see Table 3.4). It is the IP address of a respondent receiving a request or most likely an answer to a request.* IP(in-dst) *can be used to re-identify a respondent if the adversary has background knowledge allowing them to associate an IP address with a user-name. dc*2 *also contains the PSI attribute* IP(in-src)*. Similarly to* IP(out-dst)*,* IP(in-src) *indicates the IP address of a machine answering to a respondent's request that could be the domain name of a privacy-sensitive website that the respondent is visiting.*

*Data content dc*3 *and dc*2 *have two attributes in common:* IP(in-dst) *classified as QID and* IP(in-src) *classified as PSI. dc*3 *contains, in addition, the attribute* file *classified as QID because it might contain information that can be used to re-identify a respondent e.g., name and surname.* file *is also considered as a PSI attribute since files are very likely to reveal sensitive information about the respondents health, a purchase, financial information.*

Each combination of exposure and data content is considered a **privacy threat** if that data content contains PSI attributes and at least an ID attribute or a QID attribute. For example, the combination of Exposure *Exp 1* and *dc*2 (see Table 3.5) is a privacy threat (if the adversary has the necessary background information), because *dc*2 contains IP(in-src), which is a QID and IP(in-dst), which is a PSI.

If for a given combination of exposure and data content, that data content has no ID nor QID attributes or if it has no PSI attributes, that combination can be cleared as it is not a privacy threat. For example, {*Exp*1, *dc*1}, {*Exp*7, *dc*1}, and {*Exp*12, *dc*1}, highlighted in Table 3.5, are cleared. In fact *dc*1 (as shown in Table 3.6) is composed solely by IP addresses of external machines and contains no ID or QID attributes.

3.3 EPIC Fourth Step: Evaluate and prioritize privacy threat risk

In this section, we describe the fourth step of our methodology aimed at measuring the risk of each privacy threat identified in Step 3. Following a common approach in the field of IT security, we compute the privacy violation risk as the combination of likelihood of occurrence of a privacy violation L and its impact I. In the following we first describe how to measure privacy violation likelihood (Section 3.3.1), its impact (Section 3.3.2) and then we show how to measure risk (Section 3.3.3). Finally, we show how to prioritize risk mitigation actions (Section 3.3.4).

3.3.1 Privacy violation likelihood

The **privacy violation likelihood** represents the likelihood that the privacy of any respondent is violated due to the disclosure of a given data content in given data exposure. It depends on two factors: the likelihood of access (specified for each data exposure in the third step) and the likelihood that, from the exposed information, the adversary can successfully complete the privacy attack.

In order to complete a privacy attack, the adversary needs to associate the sensitive information with the respondent's identity. While in general, this association task may not be trivial, in the domain that we are considering sensitive attributes most of the time appear in data logs together with identifying or quasi-identifying information (e.g., IP, MAC address, UID). Since in this step, we are only considering data contents that contain PSI (the others have been cleared in Step 3), the likelihood of successfully completing the privacy attack corresponds to the **re-identification likelihood** i.e., the likelihood that the data respondent is re-identified.

We define this likelihood with a qualitative scale, established mainly by analyzing the ID and QID set of attributes identified in the data content in the previous step and evaluating which background knowledge the considered adversary may actually have. We provide the following guidelines and examples to assign re-identification likelihood values (c is the data content):

- *Certain.* Data respondents' identity is explicitly reported in c. Consider, for example, a company that assigns to each employee an email address in the form *name.surname* and assume that each record in c contains the senders' email address for outgoing email. In this case, each log record in c is explicitly identified.

- *High.* The adversary can discover the data respondents' identity because (i) the explicit identity is part of many records in c or (ii) c contains quasi-identifying information and the adversary has access to the background information that allows

him, with limited effort, to re-identify the respondents. As an example for case (i), consider a company in which users can choose their email addresses; c contains the senders' email address of outgoing emails. In most of the cases, the email address will be in the form *name.surname*, so the data respondent can be often identified. As an example for case (ii), consider that c contains the source IP address of outgoing HTTP connections, the adversary is the network administrator and he has background information to map an IP address to the corresponding user's name.

- *Medium.* The adversary can discover the data respondents' identity because (i) the explicit identity is seldom part of c or (ii) c contains quasi-identifying information and the adversary can use it, together with background information so that, sometimes and possibly with an effort, he can re-identify the respondent. As an example for case (i), consider that c contains the name of a file being transmitted; it is possible, though rare, that the file name contains the sender's identity like in the case of a file named *name_surname_CV*. As an example for case (ii) consider that c includes the timestamp of outgoing HTTP connection; the adversary has access to the physical entrance/exit logs for the building, so he can infer when a person was in the building, and hence, in some cases, he can find the identity of the data respondent or at least restrict the set of possible respondents to a few individuals.

- *Low.* Explicit identity is not part of c but c contains quasi-identifiers that the adversary can seldom or with a significant effort exploit to discover the respondent's identity. Consider this example: c contains the source IP address of outgoing HTTP connection. The adversary is the system administrator that, generally, does not know the association between IP addresses and employees identities. However, when a system administrator is asked for help desk support, he can become aware of a static IP address associated with a given employee, hence being able to re-identify the data respondent.

- *Negligible.* Explicit identity is not part of c and any quasi-identifying information in c, if any, can only be used to re-identify a respondent by using background information that is unlikely to be available to the adversary. Consider the case in which c contains the source IP address of outgoing HTTP connections. An external adversary does not know which user is associated with each IP address, so he cannot re-identify data respondents, especially if the address is dynamic or masked by a gateway.

The qualitative values for re-identification likelihood and likelihood of access are combined to obtain a qualitative value for the privacy violation likelihood, which is measured with a 5-values scale from *negligible* to *very-high*. Table 3.7 shows how to compute privacy violation likelihood given re-identification likelihood and the likelihood of access. The intuition behind Table 3.7 is that the two input likelihoods are combined with an operation

Table 3.7: Likelihood matrix defining privacy violation likelihood as a combination of likelihood of access and re-identification likelihood

Re-identification likelihood						
	Certain	Negligible	Medium	High	Very-High	Very-High
	High	Negligible	Low	Medium	High	Very-High
	Medium	Negligible	Low	Medium	Medium	High
	Low	Negligible	Low	Low	Low	Medium
	Negligible	Negligible	Negligible	Negligible	Negligible	Negligible
		Negligible	**Low**	**Medium**	**High**	**Authorized**
		Likelihood of Access				

similar to a product. For example, if one of the two input likelihoods is *negligible* (this is intuitively analogous to a zero probability), then the output likelihood is also *negligible*.

The **privacy violation likelihood table** (see for example Table 3.8) lists all privacy threats and for each of them it reports the *likelihood of access* (L_a) (derived from Step 3), the *re-identification likelihood* (L_{rid}), that is evaluated according to the five qualitative values defined above, the motivations behind this evaluation and, finally, the value of the *privacy violation likelihood* (L) , which is computed according to the **likelihood matrix** (see Table 3.7).

example 4. *Table 3.8 is the privacy violation likelihood table for the privacy threats identified in the running example in Section 3.2.4.*

The likelihood of access reported in this table was computed in Step 2 (see Table 3.3). Values for the re-identification likelihood were defined according to the following reasoning. Let's first consider data content dc2, including the IP addresses that an adversary can use to re-identify a respondent if he can associate it with the user-name (either directly or, for example, by first associating the IP address to the office number and then to the user-name). As observed above, security administrator *can know this association in some cases, so the re-identification likelihood is* medium. *The* network administrator *has access to the full list associating IP-addressed and user-names, so the re-identification likelihood is* high. *Finally,* external adversary *cannot associate the IP-address to the user-name, so in this case, the re-identification likelihood is* negligible.

Let's now consider data content dc3. Also, in this case, the IP-address is part of the data content, so, for each adversary, the re-identification likelihood is at least as high as with dc2. However, dc3 also contains a file (i.e., file name, file content, etc.) that can sometimes be an explicit identifier or a quasi-identifier. For the security administrator*, who is an internal adversary, the file can often identify the user. For example the* security administrator *can re-identify the user even if the file is a document signed with the first*

Table 3.8: Privacy violation likelihood (running example)

Exposure	**Data content**	L_a	L_{rid}	L
$Exp1$: DS1	$dc2$	Authorized	Medium	High
Security administrator	$dc3$	Authorized	High	Very-High
$Exp2$: C3	$dc2$	High	High	High
Network administrator	$dc3$	High	High	High
$Exp3$: P2	$dc2$	Low	Negligible	Negligible
External adversary	$dc3$	Low	Medium	Low

name only; this is possible because the security administrator *knows that there is only one person with that name, or because, from the context, the adversary recognizes the file as coming from a given office, where there is a single person with that name. For this reason, the re-identification likelihood is set to* high *for* security administrator*. Instead,* external adversary *can only re-identify the issuer when the full name is reported in the file and, in some cases, this might not even be enough, for example for very common full names. For this reason, the re-identification likelihood is set to* medium *for this adversary.*

3.3.2 Privacy violation impact severity

A privacy violation has a negative impact on the responsible organization. We model this by assigning an **impact severity** (I) value to each privacy threat. The value depends on three impact factors, defined in the following (Section 3.3.2). Impact severity can be assessed both qualitatively (Section 3.3.2) and quantitatively (Section 3.3.2).

Impact Factors.

To provide an impact severity assessment with as much accuracy as possible we first need to identify the consequences of a privacy violation, that we call *impact factors*. They are summarized in the following list:

- *Non-compliance* (I_C). If data content is exposed in a non-compliant way (e.g., respondent was not informed), then the organization might incur a certain cost in the form of e.g., non-compliance fines, respondents compensation for loss of their privacy, remediation measures to address the privacy issues that led to the unlawful leakage.
- *Failure to meet business agreements* (I_B). The organization might have agreements with end-users or other organizations that imply penalties in case of privacy violations. For example, privacy protection could be part of a service level agreement and the service provider may be subject to specific penalties in case of privacy loss.
- *Reputation Loss* (I_R). A privacy violation can have an impact on the organization reputation, that is a commercially valuable asset. Indeed, reputation loss can "erode

the ability of businesses to successfully retain their markets, maximize shareholders value, raise finance and manage debts, and remain independent" [89].

In the following, we discuss how to assign a qualitative or quantitative value to each factor. In both cases, there are three aspects that should be taken into account and that we collectively call **violation magnitude**.

i) The effect of the privacy violation on the respondent. While the effect of the privacy violation on the respondent does not have a direct impact on the organization, it is relevant for the evaluation of the three impact factors listed above. For example, if the privacy violation discloses a person's sexual orientation and this results in the person being sentenced (homosexuality is still illegal in some countries), then the reputation loss for the organization will be higher than in the case of a privacy violation that has limited impact on the data respondent.

ii) The number of respondents. It can be assessed based on the exposure magnitude (see Section 3.2.3) and an estimation of how exposed data is distributed among individuals.

iii) Nature of respondents. There are some categories whose privacy should be particularly protected (e.g., minors, social minorities) or individuals for whom a privacy violation can have worse effects than for others (e.g., a politician, a CEO).

By considering these thee aspects the expert assigns a qualitative value to the violation magnitude in the scale: *Very limited*, *Limited*, *Medium*, *Important* and *Very important*.

Impact Severity: Qualitative Assessment.

With this form of assessment a *privacy expert* and an *organization representative* jointly evaluate the severity of each impact factor for each privacy threat and assign a qualitative severity level to each factor on a 5-levels severity scale (*Low*, *Med-low*, *Med*, *Med-high* and *High*). This evaluation takes into consideration different aspects for each of the three factors. For example, the non-compliance severity will depend on the measures the organizations deployed in order to be compliant with the regulation or the lack of these measures. It also depends on the violation magnitude; indeed, in case a compensation to the violation victims is required, the non-compliance severity will scale linearly with the number of respondents affected. The reputation loss impact may depend on the adversary, on the data handled by the organization, on insufficient organizational and technical control, and most importantly by the number of individuals affected. Indeed, reputation loss is likely to scale with the privacy violation magnitude, not only in terms of number of respondents affected, but also in terms of the nature of these respondent (e.g., a privacy violation for a social minority, a celebrity or a political figure will certainly have more reputation impact than other leakages). Finally, the impact of non-fulfillment of business agreements depends on the kind of data leaked and on the agreements themselves. An example business agreement may be an SLA (service level agreement) with a cloud provider.

Table 3.9: Qualitative privacy violation impact (running example)

Exposure	**Data cont.**	**Th**	**Violation magn.**	I_C	I_B	I_R	I
*Exp*1: DS1 Sec. admin	*dc*2	*Th*1	Important	Low	Low	Low	**Low**
*Exp*7: C3 Net. admin	*dc*2	*Th*2	Very-Limited	Med-high	Low	Med-low	**Med-high**
*Exp*7: C3 Net. admin	*dc*3	*Th*3	Very-Limited	High	Low	High	**High**
*Exp*12: P2 Ext. adver.	*dc*3	*Th*4	Limited	High	Low	Med-high	**High**

SLAs usually specify a minimum level of data security and privacy. In case of failure to meet those requirements penalty fees should be paid to the client as compensation.

After evaluating the impact factors, impact severity is computed as the maximum severity level of the three factors: $I = \max(I_C, I_B, I_R)$. In fact, the five severity levels intuitively represent significantly different range of values (possibly even different orders of magnitude). Thus, the overall impact severity will most likely preserve the range of values of the highest severity among the considered impact factors.

The results are reported in the **qualitative privacy violation impact table** (see for example Table 3.9) that reports, for each privacy threat, the violation magnitude, the qualitative values of each impact factor and the resulting qualitative impact severity.

example 5. *Table 3.9 reports the qualitative privacy violation impact table for a subset of the privacy threats reported in Example 4.*

In the first row, impact severity is low*. Indeed, in threat* Th1 *users are informed that the IP addresses (both local and remote) are collected for security purposes and might be processed by the* security administrator*. For this reason, and because several measures were taken to avoid privacy violations, the non-compliance impact factor (I_C) is evaluated as* low*. I_B is low because the organization has no business agreements to fulfill. I_R is also low because the impact of this violation on reputation is minimal since a* security administrator *is somehow expected to access information about user IP addresses.*

*Impact severity of Th*2 *is med-high. Considering Th*2*, non-compliance impact factor is quite severe because respondents are not informed that the adversary can access exposed data (actually,* network administrator *is not expected to access exposed data). However, violation magnitude is very limited, because there are few respondents for the exposed data. This mitigates I_C that is evaluated as med-high. I_B is low because the organization has no business agreements. The impact on organization's reputation I is estimated med-low because the violation magnitude is* very limited *and exposed data does not contain*

particularly sensitive information (see Table 3.6 for dc2).

Threat Th3 is similar to Th2 with the difference that in this case, the adversary can also access files, which in turn can contain any type of data, including those particularly protected by existing regulations, e.g., health-related information. For this reason both I_C and I_R are high*, and consequently impact severity is also* high*.*

In threat Th4 a non-authorized person (i.e., an external adversary*) has access to cd3 that include files. Hence, similarly to Th3, I_C is* high *and consequently impact severity is* high*.*

Impact Severity: Quantitative Assessment.

Another approach to assess impact severity is to quantitatively estimate the economic cost deriving from a privacy violation. We consider the same three factors as in the qualitative approach but in this case, we associate each of them with an estimation of the economic loss.

For example, *non-compliance* cost includes: (i) the fines that the organization has to pay, (ii) the cost of remediation actions (both organizational and technical), and (iii) the compensation to pay to each affected respondent times the number of respondents.

The *reputation loss* costs are caused by the loss of trust and the degradation of the relationship between the organization and its partners, employees, investors, customers and potential future customers. It can be reflected on several levels e.g., turnover of existing customers, diminished customer acquisition, cumulative abnormal stock returns, decline of equity value [96]. It can also include the costs of efforts to control the incident disclosure and reputation repair.

The *failure to meet business agreements* cost depends on the existing business agreements and their nature.

In the case of a quantitative assessment, we compute impact severity of a privacy threat as the sum of the costs associated to each impact factor: $I = I_C + I_R + I_B$.

The results are reported in the **quantitative privacy violation impact table** that is analogous to the qualitative privacy violation impact table (Table 3.9) with the only differences that impact factors and impact severity are reported as quantitative values.

3.3.3 Privacy violation risk

As mentioned in the beginning of this section, **privacy violation risk** depends on the privacy violation likelihood and impact severity. If impact severity is assessed quantitatively, then we can compute a quantitative privacy violation risk. Otherwise, we provide a qualitative privacy violation risk assessment.

Table 3.10: Risk matrix defining qualitative privacy violation risk as a combination of privacy violation likelihood and impact severity.

Impact severity	Negligible	Low	Medium	High	Very-High
High	Low	Medium	High	High	High
Med-High	Low	Medium	Medium	High	High
Med.	Low	Low	Medium	Medium	High
Med-Low	Low	Low	Low	Medium	Medium
Low	Low	Low	Low	Low	Medium

Privacy violation likelihood

Table 3.11: Qualitative privacy violation risk

Th	**Exposure**	**Data content**	L	I	R
$Th1$	Exp1: DS1, Security admin.	$dc2$	High	Low	Low
$Th2$	Exp2: C3, Network admin.	$dc2$	High	Med-low	Medium
$Th3$	Exp2: C3, Network admin.	$dc4$	High	High	High
$Th4$	Exp3: P2, External adversary	$dc4$	Low	High	Medium

Qualitative Evaluation

We define the **qualitative privacy violation risk** with three levels: *low*, *medium* and *high*. We combine privacy violation likelihood and impact severity levels according to the **risk matrix** (see Table 3.10). The idea behind this risk matrix (in Table 3.10) is that when privacy violation likelihood is negligible, then we can exclude that the adversary can successfully complete the attack, so the risk is low. If the privacy violation likelihood is *low*, then risk is obtained by decreasing the value of the impact severity (e.g., impact severity *high* results in a *medium* risk). Similarly, if the privacy violation likelihood is *medium*, then risk is obtained by slightly decreasing the value of the impact severity (e.g., *medium-high* impact severity results in a *medium* risk but *high* impact severity results in *high* risk). A *high* value of privacy violation likelihood implies that the values of impact severity map to the same value of risk, with the exception of *medium-low* and *medium-high* that are "rounded up" to *medium* and *high* risk values, respectively. Finally, a *very-high* privacy violation likelihood results in risk values that are higher than those of the impact severity (e.g., *medium* impact severity maps to *high* risk).

The **qualitative privacy violation risk table** (see for example Table 3.11) reports, for each privacy threat, the values of privacy violation likelihood and impact severity (that were previously computed), together with the qualitative risk value that is computed based on Table 3.10.

Quantitative Evaluation

In the quantitative approach, we need to convert the qualitative measure of privacy violation likelihood into a numerical value. We propose the following association: *Very-High*= 1, *High*= 0.75, *Medium*= 0.5, *Low*= 0.25 and *Negligible*= 0. Then, for each privacy threat, we compute the quantitative privacy violation risk R as the product of the privacy violation likelihood and of impact severity: $R = L \cdot I$

The results are then reported in the **quantitative privacy violation risk table** that is analogous to the qualitative privacy violation risk table (Table 3.11) with the difference that quantitative values are reported for the privacy violation likelihood, for impact severity and risk.

3.3.4 Risk mitigation actions prioritization

In the fourth step, after assessing the risk values, we are now interested in defining in which order the privacy threats should be addressed with mitigation actions. We model this order with a **priority** value, a scale of integer values from 1 to 12 where 1 represents the highest priority.

The priority of a privacy threat depends on two factors: its privacy violation risk and the trustworthiness of the adversary involved in that privacy threat. Several definitions of trust have been proposed in the literature (see [106] for a survey). In this paper we consider the **trust** in an adversary as the organization's level of confidence about the actor not attempting to gain non-authorized data access or misusing the data to violate privacy.

This level should be assessed by taking into consideration several aspects, including legal agreements, specific training on handling personal data, personal characteristics (such as morality, skills, and behavior [67]), and organizational procedures (e.g., motivational practices and reward systems). Regarding legal agreements, note that employees with access to the system usually have to sign such agreements as part of their contract. In EPIC, the knowledge about these agreements is part of the domain knowledge acquired as input for the whole methodology (see Figure 3.1).

Human factors are receiving increasing attention in the security field. Indeed actors trust assessment is often included in risk management processes. Some approaches discuss the trust level as a part of the risk computation [151] whereas others use this level as an independent indicator to balance the risk at the decision making stage [12]. It has been observed that the first approach tends to underestimate or hide the risks involving insider threats [45]. Actually, insiders have a big potential to create threats intentionally (by attempting malicious actions) or unintentionally (through lack of experience and awareness). For this reason the EPIC methodology adopts the second approach, and we do not consider adversary trustworthiness as a factor in the evaluation of privacy violation

risk. The trustworthiness is rather used to define a priority value.

This approach has a twofold effect. On one side, it provides an effective priority classification of threats to act upon. On the other side, it provides an explicit classification of risk that also takes into account the adversaries' trustworthiness. This risk estimation will be useful in the process of deciding and designing what kind of training an actor should have in preparation to fill a high-risk position and what kind of profiles to select when hiring.

We consider the following four levels of trust.

- *Fully trusted:* Adversaries are fully trusted if they are trained to deal with personal data at the CSS level. Their activities with data are monitored by logging mechanisms and they are accountable for any personal data leakage. They often have very high privileges allowing them full access to data.
- *Trusted:* Trusted adversaries are also trained to deal with personal data and their activity is monitored. However, they have less responsibility in case of privacy leakage and have restricted access to the sensitive data.
- *Moderately trusted:* Actors are moderately trusted if they are trusted at the organization level, however, they are not specifically trained to deal with sensitive and personal information at the CSS level. These actors have often high privileges (e.g., administration privileges). They are responsible and accountable for any abuse of their privileges.
- *Untrusted:* Adversaries are considered as untrusted if they have no training on how to deal with private information and no authorizations to access the data.

We propose to use the priority distribution defined by the **priority matrix** (see Table 3.12) to combine privacy violation risk and adversary's trustworthiness in order obtain each threat priority. This matrix is designed to give more weight to the risk than to the trust. Priority of threats with the same risk level decreases (i.e., gets higher values) conversely to the trust level. In most of the cases, a privacy threat with a lower privacy violation risk than another is associated with a lower priority, with some exceptions. For example, a privacy threat with *medium* risk and *untrusted* adversary is associated with a priority higher than a privacy threat with *high* risk and *fully trusted* adversary.

The results of this procedure are reported in the **prioritized privacy threats table** (see for example Table 3.13) that indicates, for each privacy threat, its associated privacy violation risk (previously computed), the adversary trust and the resulting priority value.

Adversaries that are not fully trusted may also be at risk of sharing data with external adversaries or colluding with other adversaries. While dealing with collusion is not explicitly taken into account by EPIC, the likelihood of this scenario can be reduced by remediation actions that include specific legal obligations, and organizational measures

Table 3.12: Priority matrix defining priority as a combination of privacy violation risk and adversary trust

		Adversary trust			
		Untrusted	Moderately Trusted	Trusted	Fully Trusted
Privacy violation risk	High	1	2	3	5
	Medium	4	6	7	9
	Low	8	10	11	12

like preventing the use of personal external storage or the use of any personal device in the CSS control room.

Table 3.13: Prioritized privacy violation threats (running example).

Th.	**Exposure**	**Data content**	R	**Adversary trust**	**Priority**
$Th1$	$Exp1$: DS1, Sec. admin.	$dc2$	Low	Fully Trusted	12
$Th2$	$Exp2$: C3, Net. admin.	$dc2$	Medium	Moderately trusted	6
$Th3$	$Exp2$: C3, Net. admin.	$dc3$	High	Moderately trusted	2
$Th4$	$Exp3$: P2, Ext. adversary	$dc3$	Medium	Untrusted	4

example 6. *Table 3.13 illustrates the priority of threats considered in the previous section. The first threat* $Th1$ *has a* low *risk level. The adversary is the* security administrator *that is* fully trusted *to access and process the data content* $dc2$ *because they are highly trained to deal with personal data and assume high responsibilities for any potential leakage or misuse of this data. For these reasons, this threat has the lowest possible priority (*12*).*

In the second and third rows (i.e., $Th2$ *and* $Th3$*) the adversary is* moderately trusted. Network administrators *are trusted within the organization, but they are not authorized to access* $dc2$ *or* $dc3$ *nor specifically trained to deal with any private information collected by the CSS. Consequently,* $Th2$ *and* $Th3$ *have priority levels* 6 *and* 2*, respectively.*

In the last threat the adversary is external and hence untrusted. *Since risk is* medium*, according to Table 3.13 priority is* 4*.*

3.4 Case Study

In this section, we present a real case study used to validate our methodology. This use case considers the cybersecurity system protecting the network of an academic institution including over 15,000 hosts. We also report some of the results and findings after applying EPIC to our use case (The full analysis can be fund in [102]).

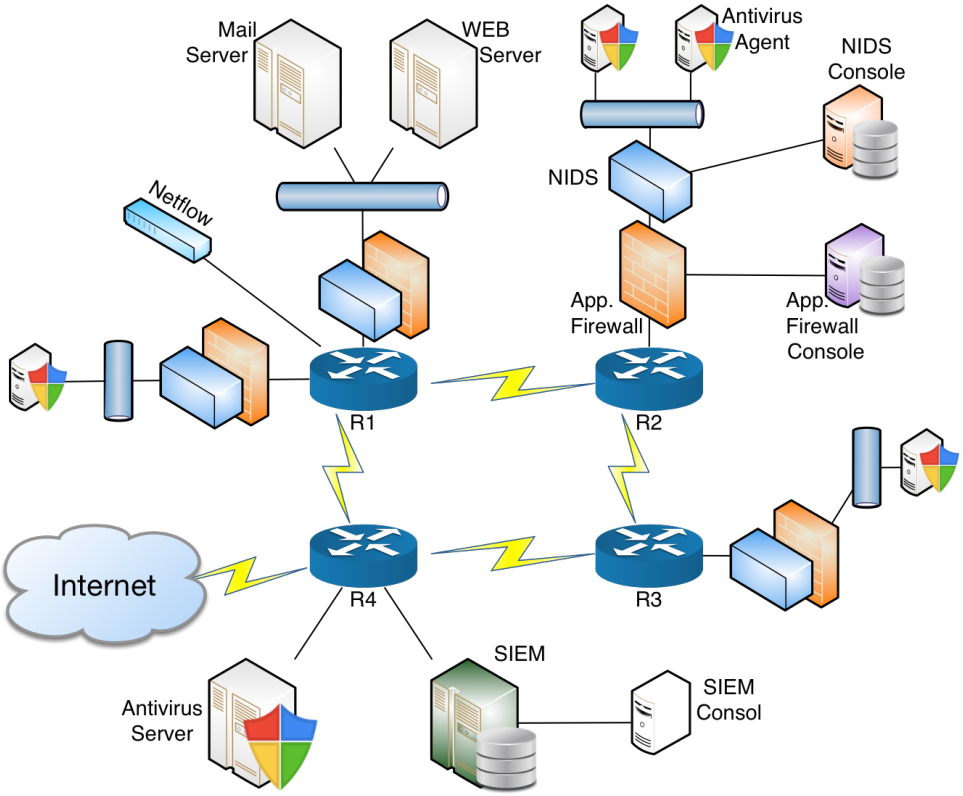


Figure 3.6: Architecture of the University's Cyber Security System

3.4.1 Case Study description

Figure 3.6 depicts the architecture of a university campus network along with its Cyber Security Systems (CCSs). From here on we will refer to the ensemble of these cybersecurity systems as *UCSS* (i.e., University Cyber Security System). The university network is divided into different network segments located in three geographic areas and connected among themselves by four main routing devices (R1, . . . , R4).

This network is protected in total by six types of cybersecurity systems: (1) Netflow network collector (in short Netflow), (2) Network Intrusion Detection Systems (NIDS), (3) application-level firewall, (4) Security Information and Event Management System (SIEM), (5) cloud antivirus and (6) security mechanisms built into the routers and in particular (i) Firewall at IP level and (ii) Virtual Private Network VPN (from now on, we will refer to this ensemble of mechanisms as router).

To identify and assess the privacy impact of UCSS, we run the four steps of the EPIC methodology for each of these six cybersecurity systems.

The Netflow is a hardware device configured to collect network packet headers. This data is mainly used to detect network anomalies e.g., performance degradation, traffic congestion, abnormal latency. In our architecture (Figure 3.6), we have a Netflow is

connected to each router and covering the whole network.

The NIDS are deployed in each network segment for monitoring the network traffic and they raise an alert when suspicious patterns are detected. Each NIDS is equipped with a remote console where alert logs are collected and a human analyst can access this data for investigation and forensic tasks

Firewall at application level is used to detect threats such as web attacks, exploitation techniques, malware infections, etc. (Figure 3.6 shows a single firewall connected to R2 but there is actually a firewall for each router). To this end, the firewall is able to process a large spectrum of data types such as executables, PDFs, emails, multimedia files etc. The firewall can be also configured to decrypt SSL traffic going to any external websites and it acts as a forward proxy. Like the other cybersecurity devices, the firewall is also equipped with a remote console to allow the security team to monitor the security events and investigate threats.

The events and threats collected by Netflow, NIDS, application level firewall and routers are sent to the SIEM for further analysis, which is considered the mastermind of UCSS. Thanks to its capabilities such as data aggregation, event correlation, and advanced forensic analytics, this system provides a view on the big picture of potential attacks running under the network and that the other CSS cannot detect separately. The SIEM has a remote console allowing interaction between the system and human agents.

The Cloud antivirus is based on a technology that uses a lightweight software component on the protected host while offloading the majority of data analysis to the antivirus provider's infrastructure. The goal of the software agent is to identify suspicious files and send them to the network cloud where multiple antiviruses and behavioral detection engines are applied simultaneously for improving detection rate. Cloud antivirus can also use a "retrospective detection" where the cloud detection engine re-scan all files already checked when a new threat is identified. Such technique can improve the detection speed.

The router as mentioned above can perform traffic filtering based on predefined networking ACLs (access control lists). ACLs indicate which traffic to allow or block based on IP addresses and port numbers. Blocked traffic can be used to investigate network attacks and incidents. Routers also allow to create and use VPNs however this security mechanism does not collect any data thus it will not be considered in our analysis.

3.4.2 Summary of the results and findings

In this section, we summarize and discuss the important results obtained after running the 4 steps methodology to the use case presented in the previous section.

In the first step we modeled the seven CSSs mentioned in Section 3.4.1 using the extended data flow diagram DFD+ (as explained in Section 3.2.2). in Figure 3.7 we show an example of DFD+ modeling the SIEM.

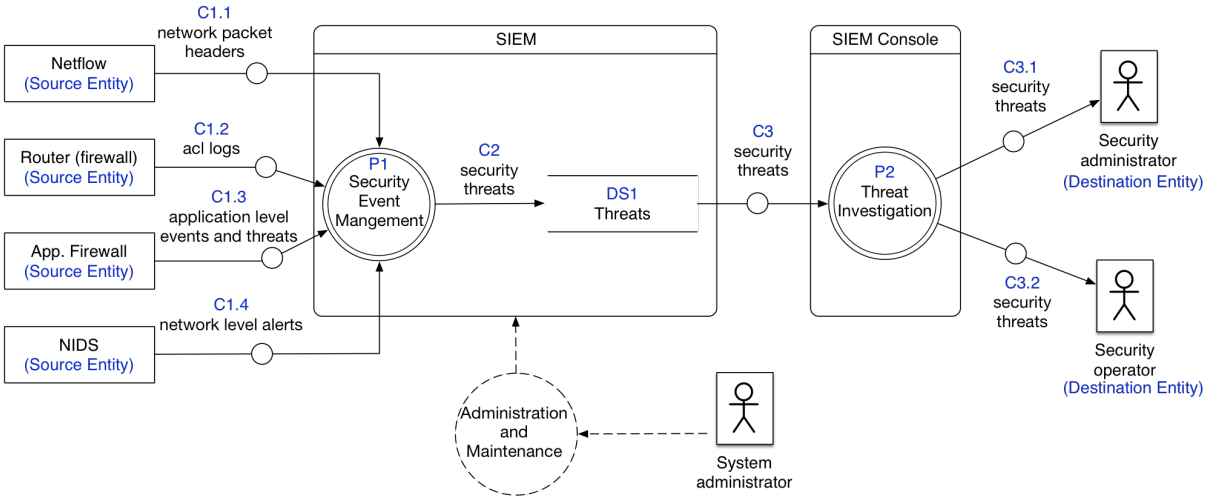


Figure 3.7: Modeling UCSS SIEM component with DFD+

In the second step, we identify 350 exposures. In Table 3.1 we report a reference list of adversaries for a cybersecurity environment that can be personalized for the specific CSS and organization being considered. In the UCSS use case, we added the following adversaries:
a) *Security operator*: in UCSS the security team is composed of a security administrator and four security operators; the operators have the same tasks but fewer privileges than the *security administrator*;
b) *Network user*: this adversary role applies to any individual with approved access to the university's network.

For each exposure, we computed the likelihood of access L_a. This assessment takes into consideration the adversaries and security mechanisms protecting each component (see Section 3.2.3). Among the identified exposures 60 had non-*negligible* likelihood of access including 38 exposures involving *authorized* actors. The rest of the exposures were cleared.

As mentioned in Section 3.2.4 the data leaked in each exposure is composed of heterogeneous types of records (i.e., records with different attributes). We call each type of records a data content.In the third step we identified 39 different data contents, composed of different combinations the attributes described in Table 3.14 [1]. We also identified 1200 privacy violation threats. In average 60% of the total number of threats involve authorized users. No threats were cleared during this step.

The analysis carried in the fourth step reveals that most authorized users threats have elevated risk levels. In Table 3.15 we report a subset of the results obtained by analyzing the SIEM (depicted in Figure 3.7).In this sample, we selected high-risk threats and we

[1]The exposures we identified actually leak other privacy neutral attributes (i.e., neither IDs, nor QIDs, nor PSIs), however for sake of brevity we don't report them in this table.

Table 3.14: UCSS attributes description

Name	Description	Domain	Example values
IP_int	IP adress (source or destination) of a machin in the local network	IP Address	192.168.100.32
IP_ext	External IP adress (source or destination)	IP Address	8.8.8.8
URL	visited sites urls and parameters if any	URL	www.sitename.com/ search?s=parameter
file_meta.	File name, size, author creation time etc.		name.pdf, 504kb, 2017-06-06 12:07:10
file	A file being transmitted	String of bytes	
email_header	Email Object, Sender and Reciver adresses	smtp header	from: to: date: subject: etc.
email_cont.	Email Object, Sender and Reciver adresses		
app. name	name of the application and protocol used	name, protocol, etc.	Thunderbird 52.1.1, smtp

can see that several threats were originated by authorized users (i.e., Security operator and Security admin). Even after taking into consideration the trustworthiness of these authorized users (acquired through training and legal commitment), the priority levels of threats evolving them is still quite important. Which means this kind of threats should not be overlooked when planning for mitigation solutions.

Table 3.15: Privacy violation risk and prioritization (SIEM)

Threat		Risk	Trust	Priority
Exposuer	Data content			
P3: Security operator	IP_int, IP_ext, URL	Medium	Trusted	7
L_a: *authorized*	IP_int, IP_ext, URL, http_content	High	Trusted	3
P3: Security admin	IP_int, IP_ext, URL	High	Fully Trusted	5
L_a: *authorized*	IP_int, IP_ext, URL, http_content	High	Fully Trusted	5
C1.3: Network admin.	IP_int, IP_ext, URL	High	Moderatly Trusted	2
L_a: *high*	IP_int, IP_ext, URL, http_content	High	Moderatly Trusted	2
DS1: Security admin. L_a: *authorized*	IP_int, IP_ext, file metadata	High	Fully Trusted	5
DS1: System admin. L_a: *mediun*	IP_int, IP_ext, file metadata	Medium	Moderatly Trusted	6

The identification and evaluation of these threats in UCSS has a high value for the academic institution, not only to better understand the privacy implications of the deployed CSS and possibly mitigate the threats but also to comply with regulation. For example, the new EU General Data Protection Regulation[2] (GDPR) requires to keep de-

[2]Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive

tailed "Records of personal data processing activities" (article 30), and the EPIC's threat analysis was an excellent tool to isolate this information for the CSS.

The prioritization is a further step with a central role in guiding the mitigation actions for the identified privacy violation threats. Considering the UCSS use case, this step of the EPIC methodology highlighted two benefits. First, it forced the trust analysis of the different actors considered as adversaries, identifying the higher reliability of security operators with respect to system and network administrators because of their different training and expertise. Second, considering only trust, *Th2* would be considered at highest priority, while considering only risk *Th1*, *Th3*, and *Th4* would be considered before *Th2*. Only the balanced evaluation of the combination of the two factors suggests the non-trivial priority order reported in Table 3.12.

3.5 Threat mitigation strategies

A natural follow-up to the EPIC analysis would be to guide through the selection and implementation of privacy protection solutions, including organizational and legal interventions. Regarding technical solutions, despite there are several privacy enhancing techniques that could be applied in this domain, a careful evaluation is required specifically for preserving data quality and computational efficiency in order not to impact on security protection. Indeed, some existing privacy enhancing techniques have been shown to reliably protect privacy, however, they often severely affect the quality of data and come with a substantial computational overhead.

Moreover, the type of selected solution strongly depends on the nature of the actors involved or originating the threat. Indeed the results of Analyses (presented in Section 3.4) allowed us to identify two categories of threats: a) unauthorized actors threats and b) authorized actors threats.

Unauthorized actors as the name indicates are actors who have no authorizations to access the data from a given components. These actors gain access by attacking the CSS of the network in which the CSS is deployed. The Unauthorized actors can be external attackers but can also be insiders i.e., companies employees, network and IT personnel etc. To address this kind of threats we need to eliminate the data leakage. This can be achieved through the reinforcement of the network security to fix the security breaches that lead to the leakage. In other cases where insiders are involved, we could think about addressing the privilege assignments and privilege abuses. Data obfuscation mechanisms such as encryption can also be used to mitigate non-authorized actors threats.

However, these solutions are not applicable to the threats originated by authorized users since these users need to access the data in order to accomplish their daily business tasks. Indeed to mitigate authorized actor threats one possible solution is to minimize the

95/46/EC (General Data Protection Regulation).

data leakage by efficiently applying the need-to-know principle when releasing the data. To do so anonymization techniques can be used to lower the granularity of identifying or sensitive information. Which will clearly lower the privacy risk but it will also impact the utility of the data and it is also expected to have a certain computational overhead. Thus this kind of solution needs to be carefully used. Another solution would be to increase our confidence the authorized actor will not miss-use the accessed data or in other words increase the actor's trust. This trust level can be increased through privacy training, the signature of legal agreements, monitoring etc.

These privacy protection measures (e.g., anonymization, trust enhancement) can be applied off-line. However in order to mitigate the negative impact, mentioned earlier, they are better applied at run-time, request by request during the access control phase where the privacy risk can be assessed more accurately by taking into consideration a number of factors, often known at run-time only, such as the access history of the requester, the security context of the query etc.

3.6 Chapter conclusions

In this chapter we presented "EPIC", a methodology to identify and evaluate privacy violation threats resulting from the deployment of an organizational cybersecurity system. The methodology guides a privacy expert, with the collaboration of the organization's security team, through four steps of analysis namely "modeling the cybersecurity system", "identifying data exposures", "identifying privacy threats" and "evaluating and prioritizing privacy threats".

We refined and validated the methodology by applying it to the actual cybersecurity system of a large academic institution. In Section 3.4 we provide a description of this CSS and briefly report the end results of the analysis (the full application of EPIC can be found in [102]).

Two contrasting needs emerged while designing the *EPIC* methodology: on one side, in order to increase the accuracy of privacy violation risk assessment a larger number of aspects needs to be modeled and deep evaluations by privacy experts need to be performed. On the other side, the methodology should be practical: the experts should be able to apply it to real systems with a reasonable effort and time. Balancing these needs required us to omit some details or special cases that add complexity to the process, while not always affecting the evaluation result in the specific context of privacy in CSS. For example, in a first attempt to model privacy violation, we explicitly took into account "linking information" i.e., attributes that can associate several pieces of information to the same individual. Consider for instance a data log that reports a given sensitive information associated with pseudo-id 123; another log contains the association between pseudo-id 123 and respondent's identity. By accessing these two logs the adversary can

violate the respondent's privacy through the "linking information" i.e., pseudo-id 123. EPIC does not explicitly provide guidance to the experts for analyzing this re-identifying method since in our case study, this form of reasoning never disclosed additional privacy threats while adding complexity. Despite we believe that our use case is representative of a large class of CSS, there may be cases that require a more detailed analysis, including linking. Actually, linking information is captured by our formal model as a special case of quasi-identifier (see our definition of quasi-identifier) and can be considered in Steps 3 and 4 of EPIC. More generally, a technically deeper analysis on specific aspects can be conducted as a second phase assessment or as part of the remediation for particular privacy threats and system components.

The privacy risk assessment resulting from the methodology can be used to compare cybersecurity systems in terms of privacy preservation. By considering the trustworthiness of the adversary together with the privacy violation risk, the methodology also provides a prioritization of the activities necessary to mitigate the risk of the identified privacy threats. This overview obtained (characteristics of the threat: adversary, component, data, risk, and priority) helps starting the elaboration of a mitigation plan.

For example, one important finding after applying the methodology to this use case (in Section 3.4) is that an important number of threats was originated by insider authorized actors. These threats have elevated privacy violation risk levels and important priorities. Privacy enhancing techniques such as encryption and security reinforcement (applied alone) can be good solutions to mitigate unauthorized access privacy threats (access achieved by unauthorized actors), but they are not applicable in case of authorized access. Therefore other solutions need to be proposed to address these type of threats mainly by decreasing the amount of data accessed by authorized actors. This kind of solutions, e.g., anonymization, impacts the quality of data thus the level of anonymization should be decided carefully depending on the privacy risk.

In this optic, we propose a privacy-aware risk-based access control system, where the privacy risk is assessed at run-time for each request. If this risk is higher than the request's trust, several strategies can be applied to lower the risk and/or enhance the trust. In the remaining chapters of this thesis, we will provide more details about this approach.

Chapter 4

Trust- and Risk-based access control

Access control mechanisms are fundamental mechanisms in computer security used to ensure that only authenticated and authorized users can perform allowed actions on given resources under given circumstances. The rapid evolution and diversification in today's ICT landscape brought various challenges for access control. Indeed this new environment requires a high level of data availability and calls for more flexible access evaluation. In fact, in commonly used access control systems (e.g., RBAC, MAC), it is current practice to grant all-or-nothing access. Although this approach supports privacy and confidentiality, it lacks flexibility and limits data exploitation and availability. Vigorous studies have been conducted to dress these challenges. Among the proposed solutions risk-aware access control approaches received increasing attention during the last years. These systems grant or deny access to resources based on the notion of risk. It has many advantages compared to classic *approaches, allowing for more flexibility, and ultimately supporting for a better exploitation of data.*

In this chapter we propose a novel trust- and risk-based access control framework *supporting run-time trust and risk assessment. For each request, access is evaluated based on a combination of these trust and risk values. Differently, from existing models, our framework supports access control in dynamic contexts through trust enhancement mechanisms and risk mitigation strategies. This allows striking a balance between the risk associated with a data request and the trustworthiness of the requester. If the risk is too large compared to the trust level, then the framework can identify adaptive strategies leading to a decrease of the risk (e.g., by removing/obfuscation part of the data through anonymization) or to increase the trust level (e.g., by asking for additional obligations to the requester). We outline a modular architecture to realize our model, and we describe how these strategies can be actually realized in a realistic use case.*

4.1 Introduction

The increase in the amount of data generated by today's digital society is astonishing. According to IDC estimate [71], the global volume of digital data will double every two years, reaching 44 trillion gigabytes by 2020. The availability of such large and diverse datasets (*big data*) enables the creation of new data-based businesses or optimizing existing process (real-time customization, predictive analytics, etc.).

Yet, organization and companies are often unable to exploit the full potential of this data (e.g., providing access to analysts, sharing with and accessing partners data). Indeed this data often contain confidential and sensitive information and providing access to this information carries multiple risks of intentional or accidental misuse [1]. Moreover currently used access-control mechanisms have major limitations for handling complex data sharing scenario while managing potential security and legal risks. Already few years ago, the JASON MITRE report [121] indicated that the inflexibility of (still-used) access control mechanisms is a major obstacle when dealing with diverse data sources in a dynamic environment. Therefore the success key of any organization is to find the right balance between providing *Flexibility* while providing essential information and ensuring the confidentiality of the data.

To overcome this problem, access control systems must weigh the risks of the incoming requests. Then access decisions must be based on an estimation of expected cost and benefits, and not (as in traditional access control systems) on a predefined policy that statically defines what accesses are allowed and denied. In such approaches, referred to as risk-based access control, for each access request, the corresponding risk is estimated and if the risk is less than a threshold then access is guaranteed, otherwise, it is denied. Although existing risk-based access control models provide an important step towards a better management and exploitation of data, they have a number of drawbacks which limit their effectiveness. In particular, most of the existing risk-based systems only support binary access decisions: the outcome is *allowed* or *denied*. Whereas in real-life we often need to handle exceptions based on additional conditions. For instance, the access to sensitive medical data (e.g., by a non-treating doctor) should be allowed in a situation of emergency in return the doctor should sign non-disclosure agreements. In other words, if the risk is higher than a certain threshold, the system should be able to propose appropriate risk mitigation measures instead of denting any risky access. This way it enhances the exploitation of the data while maintaining an acceptable risk level.

In this Chapter, we propose a novel access control framework that combines trust with risk and supports access control in dynamic contexts through. This allows us to strike a balance between the risk and the trustworthiness when evaluating a data request. If the risk is too large compared to the trust level, then the framework can identify adaptive adjustment strategies that can decrease the risk (e.g., by removing part of the data)

and/or increase the trust level (e.g., by asking for additional obligations to the requester) to increase the exploitability of the data instead of denying every "risky" request.

Our framework enjoys a number of features:

1. it explicitly models trust and risk, which are the key factors of any business decision;
2. it increases the flexibility of existing risk-based access control, by introducing trust.
3. it provides an understandable way to implement and enforce risk and trust adjustment operations both prior to and after issuing the access decision;
4. it supports complex authorization scenarios by simply changing the configuration (trust and risk configuration modules, and corresponding mitigation/enhancement strategies).
5. it can be realized using a declarative policy language with a number of advantages including usability, flexibility, and scalability. As we will see in Section 4.4 the architecture and the policy structure[1] we propose can be readily implemented as an extension of a well-known declarative authorization language XACML.

With motivating use case, we will illustrate how the framework can work in practice, addressing access control requirements in a natural way, that would otherwise need complex authorization structure and calibration.

In the next section (Section 4.2) we describe the selected use case. In Section 4.3 we introduce our trust- and risk-based access control model and discuss approaches to trust and risk evaluation and adjustment. In Section 4.4 we provide an architectural view of our access control framework. In Section 4.6 we show how the proposed framework addresses the requirements of our use case, and we conclude in Section 4.7 with some final remarks.

4.2 Use Case

Consider a company with an ERP system with a Human Resource (HR) Management module, enabled with the proposed trust and risk-based access control system (see Figure 4.1). By using the ERP functionalities corporate user Alice can generate an HR report containing a list of employees with their location and salaries. The report contains sensitive information and personal data, typically subject to strict regulations, such as European Directive on Data Protection 95/46/EC that, among other terms, prevents European citizens personal information to be transferred outside EU countries (with some exceptions that we do not consider here). and the company has strict rules for accessing the data such as security measures to minimize the disclosure risk when data are moved

[1] In Chapter5 Section 5.7 we will present a sample of XACML attribute-based policies implementing privacy-aware risk-based authorizations described in 4.1

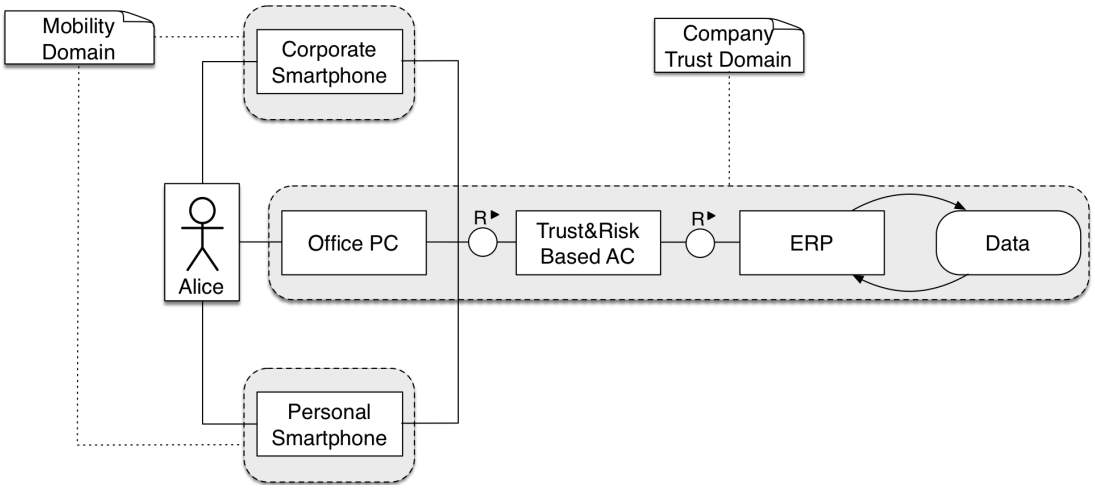


Figure 4.1: Use Case: Alice accessing an HR report with personal data covered by EU Directive on Data Protection 95/46/EC.

outside the company. The risk scenario considered is the leakage of the salary information associated with a specific employee (re-identification risk). To ensure compliance with EU data protection laws, additional restrictions must be applied if data are accessed outside EU. Data Controllers (i.e., the entities responsible for the personal data collection and management processes, in this case, the company) have the legal obligation to adhere to the directive but at the same time, Alice must be able to access the report.

In her daily business, Alice may access the report using multiple devices: her office PC at corporate premises, a corporate smartphone, and her own smartphone. Access, in mobility, suffers from a high level of risk, since it is more exposed to external attacks and, depending on the geographical location, different rules may apply. A conservative approach, easily implementable with traditional access control systems, would imply a security policy like that:

- if Alice is on premises, then access is granted
- if Alice is in mobility, access is denied as the security and compliance risks could be too high

Basically, access is limited to corporate premises, where full data can be viewed whereas outside no information is available and no reports can be produced. Even though this approach could seem simplistic, many real-life access control systems offer a similar level of functionality [153].

Ideally, Alice would like to get a wider access to the data, and perform her business tasks (e.g., reporting) also in mobility, using different devices in multiple locations, but still keeping security risk under control, as summarized in Table 4.1.

In the next sections, In the next sections, we will show how these scenarios can be

Table 4.1: Possible usage scenarios, comprising different devices and locations, and expected utility (i.e., type of reports needed) and security levels

	Scenario			Expected	
#	Device	Location	Administration	Utility	Security
1	PC	on premises	corporate	full access	no restriction
2	Smartphone	EU	corporate	grouped by country	medium risk
3	Smartphone	EU	personal	grouped by region	minimal risk
4	Smartphone	no EU	-	no access	no access

realized in our framework. We will present our trust- and risk-based access control model and show how these scenarios can be realized in our framework.

4.3 Trust- and Risk-based access control model

In this section, we provide a general description of our trust- and risk-based access control model. We will also discuss the risk and trust models, assessment and adjustment strategies.

4.3.1 Risk-based authorization model

The framework evaluates access decisions using the trust and risk values associated with the access request. An *access request* issued by subject u to carry out action a (e.g., read or write) on resource obj (e.g., a file) in context C is modeled as a quadruple $req = (u, a, obj, C)$. For instance, a request issued by user Alice to read file *HR-report.xlsx* from her corporate cell phone during her presence in the company's premises is represented by $req_0 = (\text{Alice}, \text{read}, \text{HR-report.xlsx}, \text{corporate cell phone}, \text{on premises})$. Let Π be the access control policy of the organization. We write $\Pi(req) = \mathsf{granted}$ to denote that req is granted access by Π, and $\Pi(req) = \mathsf{denied}$ to denote that req is *explicitly* denied by Π. Some access control models, do not explicitly deny access in these models $\Pi(req) \neq \mathsf{granted} \equiv \Pi(req) = \mathsf{denied}$. However this approach is clearly more restrictive, and since we would to gain as much flexibility as possible we will only use this formulation ($Auth_\Pi(req) = \mathbf{deny}$ if $\Pi(req) \neq \mathsf{granted}$) when the access control does not allow otherwise.

$$Auth_\Pi(req) = \begin{cases} \mathbf{deny} & \text{if } \Pi(req) = \mathsf{denied} \\ \sigma(req) & \text{if } T(req) - R(req) < 0 \\ \mathbf{grant} & \text{otherwise} \end{cases} \tag{4.1}$$

Policy Π can be extended so to take into account the risk $R(req)$ and the trustworthiness $T(req)$ of the request req as shown in Eq. 4.1 above If the authorization is not *denied* by Π, the request is evaluated by comparing the risk $R(req)$ with the trustworthiness $T(req)$. $T(req)$ plays the role of a risk threshold (in practice, the maximum amount of risk that a requester can take in a certain context). If $T \geq R$ access is granted, otherwise it cannot be granted *as is*. In the latter case, instead of denying access, the system may identify and propose an *adjustment strategy* σ whose application meets the condition $T \geq R$. Adjustment strategies can be either *(i) risk mitigating*, i.e. mitigation strategies whose application decreases the risk R or *(ii) trust enhancing*, i.e. mitigation strategies whose application increases the trustworthiness T. An example of risk mitigating strategies the imposition of obligations of the handling of data e.g., retention period restricted to 2 hours. An example of trust enhancing strategies are the (temporary) privilege escalation and provision of an additional, stronger proofs of identity (e.g. a two-factor authentication). In the next chapter (Chapter 5) we will present policy sample and describe how different elements of Eq. 4.1 are implemented in the policy.

4.3.2 Modeling Trust

Trust is a wide concept, and different definitions have been proposed in literature [76]. To our scope, we can use the definition by McKnight and Chervany [105], which better related to the concepts of utility and risk attitude. [2]

> *Trust is the extent to which one party is willing to depend on something or somebody in a given situation with a feeling of relative security, even though negative consequences are possible.*

In our case, we consider trust expressing the level of confidence the resource controller has that a user u will not misuse the resource he/she wants to access. We expect this level to depend on the user u (identity, role, and previous behavior) and in the given context C (e.g., the device or system environment he is using).

Trust values are assigned in various ways depending on the specific use cases. For example in reputation models, trust assessments from other entities are combined to compose a trust evaluation, or in behavioral trust, a value is assigned based on the historical records of transactions [76]. Trust can be also derived from assessing a set of *trust indicators* such as security metrics (e.g., level of authentication) and from trust assertions (e.g. stamp of approval) issued by trusted entities (i.e., certification authorities).

From the risk-based system point of view, the identity of the requester heavily depends on the effectiveness of the authentication mechanism employed. To take into account this, the trustworthiness of user u in context C, say $T_{\mathit{eff}}(u, C)$, should take into account the

[2] A popular used definition is from Gambetta [61], which stresses the *reliability* aspects of trust. For a discussion see [76].

possibility that the authentication is not carried out correctly (e.g., an identity theft scenario). This situation can be modeled in our framework by replacing $T(u, C)$ with $T_{eff}(u, C)$ in Eq. 4.1, where

$$T_{eff}(u) = T(u)(1 - P_{it}) + T(u' \neq u)P_{it} \tag{4.2}$$

where $T(u' \neq u)$ is the Trust associated to any, not specified, other user that is not u, in practice it should be zero or negligible and P_{it} is the probability of an identity theft. P_{it} represents the strength of the authentication mechanisms.

4.3.3 Modeling Risk

Risk is defined by the likelihood and the impact of the occurrence of one or more a series of failure scenarios $s \in S$ (also called risk scenarios). Although different quantitative risk methodologies exist, see [31] and references therein, for independent scenarios as risk can be computed by:

$$R(obj, p) = \sum_{s \in S(C)} P(s)I(s)$$

where S is the set of possible failure scenarios related to the access of p in the context C, $P(s)$ is the probability of occurrence of the failure scenario s, and $I(s)$ the associated impact (often measured as monetary cost).

The risk exposure can be decreased implementing a set of controls and mechanisms, and in this case, we refer it as residual risk. In addition, temporary risk mitigation strategies can be applied to further reduce the risk. In case of access control, they include for example, decreasing the probability of failure, by obfuscating (part of) the data (e.g., anonymization) or imposing usage control restrictions (e.g., data retention period); or decreasing the impact, by insurance.

Eq. 4.1 implies that trust and risk are measured in the same units. Ideally, risk should be measured in monetary units (since the impact is the cost of occurrence of a certain scenario), and, accordingly, trust should have the same units, as in the previous example for financial transactions. Unfortunately, estimating risk in information systems is much less consolidated practice, due to: i) the limited availability of historical data on failure scenarios, which makes difficult to estimate the corresponding probabilities. ii) the difficulty to estimate the impact of a failure to protect intangible digital assets.[3]

To overcome these problems, existing risk-based access control systems use various approaches: they estimate these values from the parameters of traditional (non-risk based) access control models (e.g., see [37] for multi-level security models), they use relative

[3]For these reasons, so far, most of the risk assessments for information system are qualitative, where probability and impacts are classified in broad categories and no explicit numerical values are assigned (e.g., in many application of ISO 27005:2011 [74]).

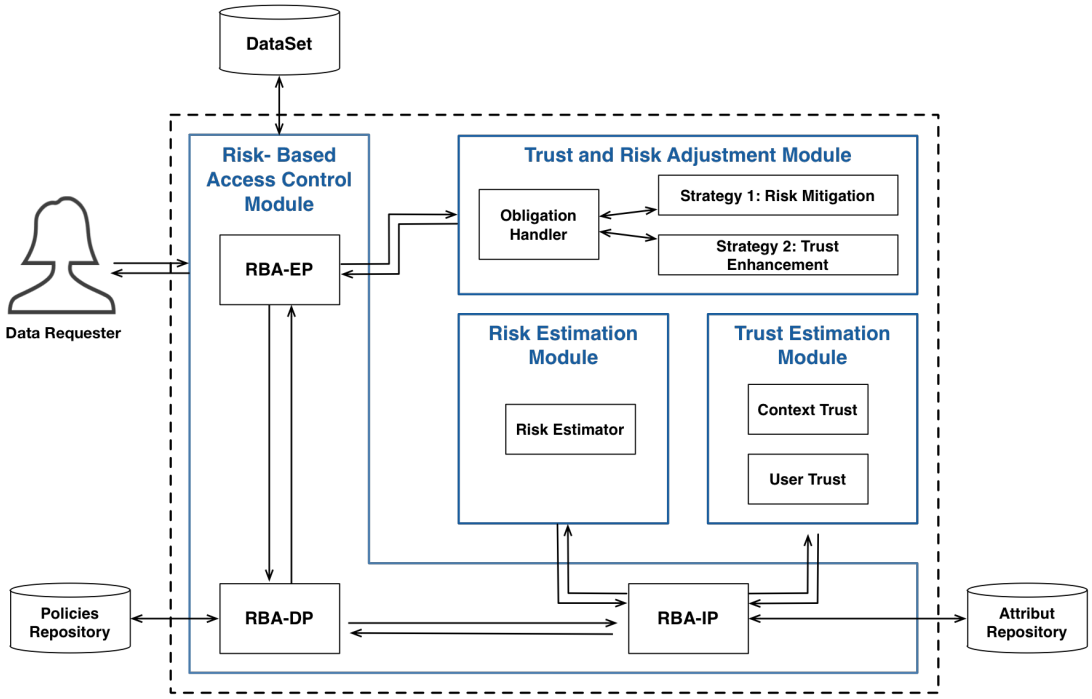


Figure 4.2: Architecture of the trust- and risk-based privacy-aware access control framework.

measures for both trust and risk (in practice they normalize these quantities in the interval $[0, 1]$, see [13]), or they use heuristics for estimating these numbers from qualitative risk assessments [31].

In the sequel, to demonstrate our approach, we will consider a single risk factor related to data privacy (re-identification risk). This allows us to compare trust, normalized in the interval $[0, 1]$, directly with the probability of the risk scenario. The model can clearly include any other security risk factors, as far as a quantitative risk estimation is possible, for example, deriving risk values from the rating of the Common Vulnerability Scoring System (CVSS) [68].

4.4 Architecture

In this section, we present an abstract architecture for our trust- and risk-based access control framework, and we explain the different steps of the data request evaluation workflow. Figure 4.2 depicts the four main modules of our framework:

Risk-Based Access Control Module. This module is the entry point of the framework. It intercepts each data request to perform the access evaluation. Access will be fully grant, partial/conditionally grant, or denied (see Eq. 4.1) following a risk-based approach were the request risk and trustworthiness are assessed and compared,

possibly after applying the adequate adjustment strategy (set of operations aiming to lower risk and/or enhance trust).

This module is composed of three main components inspired by the XACML standard reference architecture[4](PEP, PEP, and PIP), which we adapted to the risk-based authorization model that requires more complex operations, such as risk and trust assessment and adjustment. We call the modified components respectively RBA-EP (Risk-Based Authorization - Evaluation Point), RBA-DP (Risk-Based Authorization - Decision Point), and (Risk-Based Authorization - Information Point).

Risk Estimation module. This module is used to assess the level of risk, based on the data requested, context and criteria defined in the risk estimator configuration. To estimate risk, this module can require additional information about requester and context from the RBA-IP.

Trust Estimation module. This module is used to assess the trust level of a request. In particular based on user attributes like role, and past behavior. Trust computation can also take into account context attributes, for instance, in our case, access context (purposes) e.g., access is requested for maintenance of a pattern, and security context e.g., access is requested during a security alert.

Trust and Risk Adjustment module. This module is activated by the Risk-Based Access Control module (more precisely by the RBA-EP) to adjust risk and/or trust levels, when the access risk to the requested resource exceeds the trust level, in such a case, two possible options are available:

- *Decrease Risk:* if this option is selected this module, first, produces an estimation of the minimal transformation level to be applied in order to meet the required risk level (e.g., the minimal level of obfuscation, generalization). Then, the optimal risk mitigation operations are applied (e.g., access restrictions and usage control operations, which decrease risk but minimize the information loss).
- *Enhance Trust:* the trustworthiness estimation can be increased in return for the execution of certain operations. Before granting access to the resource (e.g., second-factor authentication) at the moment of the access (e.g., monitoring or notifications) or when specific events occur after granting access; for example, in

[4]In the XACML3.0 (eXtensible Access Control Markup Language) standard [58] the PDP is the point that evaluates an access request against an authorization policy and issues an access decision and the PEP Policy Enforcement Point is the point that intercepts user's request call the PDP for an access decision then enforce this decision by allowing or denying the access. The PIP is the point that can be called to provide additional information about the resource, requester or environment.

usage control, we may prescribe the deletion of a resource after that a retention period expires.

In Figure 4.3, we illustrate the interactions between different modules of the framework during the request evaluation and decision enforcement: First, the Risk-Based Access Control module, more precisely the RBA-EP, extracts the request target (i.e., resource, subject/requester and environment/context attributes) and sends the information to the RBA-DP for evaluation. The RBA-DP checks if there are any access policies/rules matching this target in the Policy Repository. During the matching, the RBA-IP can be called to resolve or provide more information about some attribute. If the matching fails (e.g., missing or unknown attributes) or if the target matches a policy denying the access to the specific target (Blacklisted Target), a *deny* response is sent to the requester. If a match is found and the policy does not explicitly deny access, we need to compare the risk and trust levels to check if we should *grant* access or if we need to apply adjustments to these levels (see 4.3.1 (Eq.4.1)). To this aim, the Risk-Based Access Control module calls the Risk Estimation Module to determine the risk level of the request and the Trust Estimation Module to determine the requester and context trust. Then, Trust and Risk Mitigation Module enters into play to increase trust and/or reduce risk, if necessary, before granting partial or conditional access to the resource. In the next chapter (Chapter 5) we will provide a sample of risk-based policy and more description of the evaluation process in the context of privacy.

4.5 Trust and Risk adjustment Strategies.

4.5.1 Risk Mitigation.

Risk can be mitigated by decreasing/limiting access to the resource. A possible way is to limit the quantity of data accessed e.g., by removing confidential data. In the context of data privacy, anonymization (see [41]) is a commonly used practice to reduce privacy risks, by obfuscating partially or completely, the personally identifiable information in a dataset. Other techniques can be used to lower the sensitivity of the data by increasing the granularity of the sensitive information. We will further discuss these techniques in the next chapter (Chapter 5, where we will focus on privacy risks.

4.5.2 Trust Enhancement.

Enhancing the trust results in raising the risk threshold and adopting a more permissive evaluation. In return, proofs and/or guarantees limiting possible misuse scenarios must be provided; for instance by asking the user to provide a stronger authentication to limit the likelihood of an identity theft and temporarily increase the trust of a user T_{user}, which

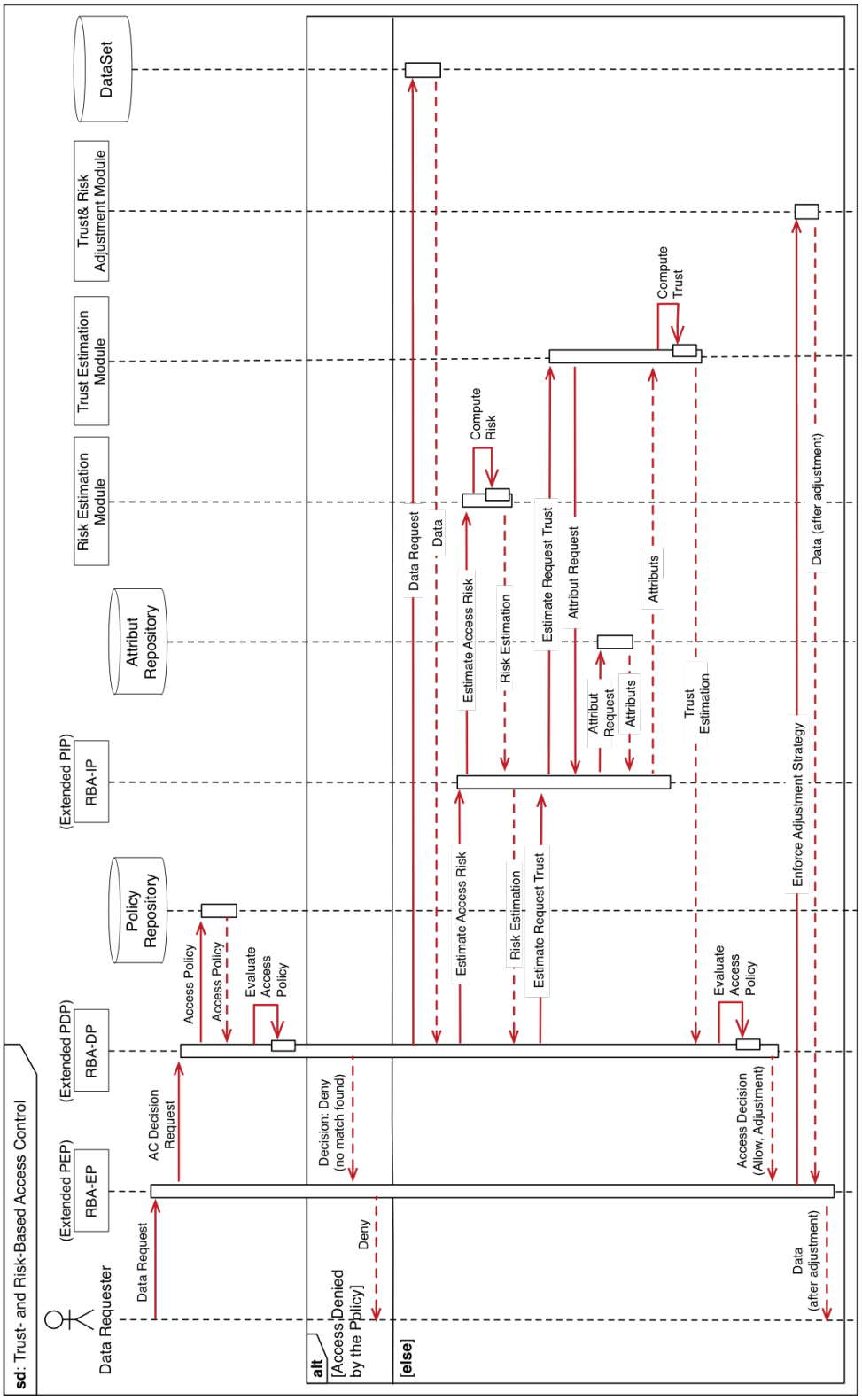


Figure 4.3: Sequence of interactions in the trust- and risk-based access control framework.

Table 4.2: Trust values in different contexts C

Context	$T(u, C)$
C_1: On premise	1.0
C_2: Mobility (secure)	0.5
C_3: Mobility (standard)	0.1
C_4: Mobility (outside EU)	0.0

impacts the trust value according to Eq. 4.2. We can also provide restricted access to a resource for a determined amount of time, then delete the resource (data), this represents a change in the context and, accordingly, it increases $T_{context}$ impacting the request trust value as well.

4.5.3 Trust and risk adjustment by obligation.

Trust and risk adjustment strategies can be implemented in the form of access and usage control obligations.

Obligations are actions or operations that must be carried out as result of an authorization decision. In the standard XACML architecture [115], obligations are defined as parts of policies and included in authorization responses created by the PDP; they are enforced by the PEP on behalf of the subject issuing the authorization request. Besides their application as an outcome of the authorization decisions, obligations may also be applied during or after the consumption of a requested resource or the execution of a requested operation [2, 114]: for example, a policy may state a specific retention period for any copy of a resource whose access was granted to the requester. In these cases, a trusted component must exist that is able to operate in real time as a PEP. This situation is generally referred as *Usage Control* (UC) [136]. UC models and mechanisms have been proposed to address confidentiality and privacy requirements [5], and applied to both the cloud and the mobile environments [51, 129].

AC/UC policy definitions may comprise a broader set of directives, regulating runtime aspects originated from an authorized access; for example, a policy may prescribe to monitor the location where a mobile user consumes a resource and to react with a deletion obligation in case the user leaves the country. Such capabilities are particularly useful to achieve compliance with directives (law requirements or corporate policies): for example, data privacy regulations introduced in Section 4.5 impose the application of certain principles and UC can enforce automatically some aspects [155].

Therefore, the usage of obligations, when their enforcement is guaranteed, can be considered as a means to enhance a request's trust estimation in our proposed system.

In fact, it can be assumed that prescriptions specified by a security policy are applied and that they can regulate how resources or operations are used, thus ensuring their compliance. For instance, in our use case (Section 4.2) the trust level could change with the context as shown in Table 4.2. For the sake of simplicity we assume that trust is independent from the specific user, i.e., $T(u, C) = T(u', C)$ for all contexts C and users u and u'. In the for the most trusted environment (On-premise) we can thus have $T(u, C_1) = 1$. C_2 refers a context of mobility inside the EU territories where the request is issued from a secure device (e.g., a corporate smartphone or laptop). In addition, in this environment, potential UC/AC obligations enforcement can be guaranteed. This context has a trust level $T(u, C_1) = 0,5$. In C_2 access is requested in mobility within EU from a non-certified device (e.g., a personal smartphone). The security level of this device is not verified and UC/AC obligations enforcement cannot be guaranteed. Therefore $T(u, C_1) = 0.1$ whereas for requests coming from outside the EU that cannot be trusted and thus $T(u, C_4) = 0$.

4.6 Application to the use case

We now show how our framework can support the scenarios introduced in Section 4.2 and achieve the expected utility and security levels (see Table 4.1). In all scenarios considered, we assume user Alice requests access to the data-view v presented in Table 4.3.

Table 4.3: HR report: original view

Name	Job	Location	**Salary**
Timothy Lulic	Senior Developer	London	74200
Alice Salamon	Support	London	45000
Perry Coda	Junior Developer	London	52000
Tom Torreira	Admin	Milan	28000
Ron Savic	Senior Developer	Rome	66000
Omer Regini	Senior Developer	Shanghai	47000
Bob Eramo	Support	Macau	18000
Amber Mesb	Admin	Bangalore	30000
Elise Moisander	Admin	Bangalore	31000

We will now describe the evaluation results of this access request $req(Alice, read, v, C)$ in four different contexts $C \in \{C_1, C_2, C_3, C_4\}$ introduced in our use case (Section 4.2). In Table 4.4 we report the initial request trust and risk ($T \equiv T_C$ and $R \equiv R(v, read, C)$) for each of the four scenarios (i.e., each of the foure contexts C). we also report the adjustment strategies applied in each scenario, the trust and risk values after adjustment ($T' \equiv T'_C$ and $R' \equiv R'(v, read, C)$), and finally the access decision.

Table 4.4: Trust and Risk adjustment strategies applied to the request $req(Alice, read, v, C)$

Context	T	R	Adjustment	T'	R'	Decision
C_1	1.0	1	None needed	-	-	**Allow**
C_2	0.5	1	Prevent sharing, Delete after 2 hours, Generalize to country level	0.75	0.75	**Allow**
C_3	0.1	1	Generalize to regional level Decrease the sensitivity	0.1	0.1	**Allow**
C_4	0.0	1	None possible	-	-	**Deny**

Scenario #1: Access from business environment. In the first scenario, Alice asks for the HR report from a business environment. The Risk Estimation module is called to estimate the access risk associated with the request $req_1 = (Alice, read, v, C_1)$ set: $R(v, read, C_1) = 1$, since the report contains personal data with an elevate likelihood of re-identifying individuals as well as learning a sensitive information (i.e., salary) about them. Alice is a trusted actor and she authorized by a security policy to perform this operation, therefore we will consider her from here on, for simplicity sake, we will consider her fully trusted and consider the context trust as request trust $T(Alice, C) = T_C$. The Trust Estimation module, in turn, computes the trust associated to the context where the request is originated: $T_{C_1} = 1$, since Alice is in her office. Therefore, $Auth(req_1) =$ **Allow** and access is granted with no need to apply any adjustment operation.

Scenario #2: Access, in mobility, from EU using corporate smartphone. Since the request is performed in mobility $T_{C_2} = 0.5$ and while $R(v, read, C_2) = 1$. The Trust and Risk Adjustment module then triggers the trust enhancement and risk mitigation strategies. Specific AC/UC obligations are thus assigned to the report (e.g., do not share, delete after 2 hours, only usable in EU) to be enforced by an obligation enforcement engine deployed on the corporate smartphone. The application of these measures increases the trust in the context to $T'_{C_2} = 0.75$. To decrease risk, generalizing the report to country level (by obfuscating the identity of the employees) allows to reduce the re-identification risk to 0.75[5]. Therefore, $Auth(req'_2) =$ **Allow** ($req'_2 = (Alice, read, v_2, C_2)$) and Alice receives the generalized view (v_2) of Table 4.5.

Scenario #3: Access, in mobility, from EU using personal smartphone. This scenario is similar to the previous one, with the notable exception that now no trust enhancing measures can be enforced on the mobile phone. Therefore, the Trust and Risk Adjustment

[5]risk in this example was assessed based on the re-identification risk and the sensitivity of the information that will be further explained in Chapter 5, in the next chapter (Chapter 5) we will provide more details about how we compute this risk using well-known privacy metrics such as k-anonymity

Table 4.5: HR report v_2: anonymized (country level).

Name	Job	Location	**Salary**
***	***	UK	74200
***	***	UK	45000
***	***	Italy	52000
***	***	Italy	28000
***	***	Italy	66000
***	***	China	47000
***	***	China	18000
***	***	India	30000
***	***	India	31000

module can only apply the risk mitigation strategy. By generalizing the report to regional level and decreasing the granularity/sensitivity of the salaries to salary ranges the risk is mitigated to 0.1. The trust of the request considering the context C_3, is $T_{C_3} = 0.1$ (see Table 4.1 access though personal smartphone), and thus after adjustment $Auth(req'_3) =$ **Allow** $(req'_3 = (Alice, read, v_3, C_3))$. The report (data view v_3) received by Alice in this scenario is given in Table 4.6.

Table 4.6: HR report v_3: anonymized (region level) and salary ranges.

Name	Job	Location	**Salary**
***	***	EMEA	[71k-90k]
***	***	EMEA	[31k-50k]
***	***	EMEA	[51k-70k]
***	***	EMEA	[10k-30k]
***	***	EMEA	[51k-70k]
***	***	APAC	[31k-50k]
***	***	APAC	[10k-30k]
***	***	APAC	[10k-30k]
***	***	APAC	[31k-50k]

Scenario #4: Access in mobility from outside EU with personal smartphone. In this case, the risk of violating the regulations is maximum. This means that the trust in the environment is 0, no mitigation strategies may be adopted and therefore $T_{C_4} = 0$ (request from outside EU), $R(v, read, C_3) = 1$, and thus for the request $req_4 = (Alice, read, v, C_4)$ the decision is $Auth(req_4) =$ **Deny**.

4.7 Chapter conclusions

Motivated by the need to balance the advantages of big data availability, and stringent security and confidentiality requirements, novel access control paradigms are emerging. Risk plays a central role, and access control decisions can mimic the business decision process, where risk is assessed relatively to trust. In this chapter, we proposed a novel access control framework based on these two factors (trust and risk) and showed that it can address complex authorization requirements by dynamically applying strategies for risk mitigation and trust enhancement. The possibility to play with both risk and trust at the same time and its application to a real use case are the main novelties of our work.

Our approach, although promising still faces a number of open issues. In particular, the overall approach (as for any quantitative risk model) relies on the numerical estimation of risk and trust which are difficult to compute [16, 111]. Indeed, the diversity of risk scenarios, the intangible nature of trust, and the limited amount of historical data for incidents makes an accurate quantitative assessment extremely difficult.

In The next chapter, we will adapt our risk-based access control approach to the context of data privacy. Moreover, we will propose to measure the trust and risk values using some domain-specific heuristics. Using these heuristics we will show how it is possible to derive sound relative estimation (i.e., using dimensionless units) for trust and risk (for a specific usage scenario).

Chapter 5

Privacy-aware risk-based access control

Several risk-based access control models were recently proposed to address flexibility issues in traditional access control models. However, very little attention was given to privacy. For instance, if we take the case of querying databases containing personal and sensitive information, the current practice is to adopt restrictive approaches, reluctant to take any risks, to avoid the disclosure of any sensitive information. These approaches ensure privacy but they offer very limited access to the data, which does not fit any more in the new data-driven environment where companies and organizations evolve.

These issues can be addressed using the risk-based access control approach we propose in the previous chapter. In fact our model enhances data exploitation by adopting a flexible risk management, however, our model (more specifically the risk assessment model and risk and trust adjustment strategies) needs to be adapted to the context of data privacy.

*In this privacy-aware model, we propose to assess the privacy risk using well-known privacy metrics. Indeed there are two main categories of approaches to assess and handle data privacy in the literature, the "*syntactic approaches*" and "*differential privacy*". Each category has several advantages and a number of issues, but both categories are equally interesting for us since they cover two different areas. Syntactic metrics are more suited in the context of Privacy-Preserving Data Publishing (PPDP) whereas differential privacy is typically used for Privacy-Preserving Data Mining (PPDM).*

In this chapter, we will focus on a privacy-aware risk-based access control model using syntactic anonymity for privacy risk assessment and mitigation (a model using differential privacy will be discussed in another chapter). We will design

two frameworks following this model, the first implemented on top of a role-based access control model (RBAC) and the second using the attribute-based access control (ABAC). Then we will show how the two frameworks can simultaneously address both the privacy and the utility requirements in two different industrial case studies. The experimental results presented at the end of this chapter prove that this approach leads to meaningful results, and real-time performance, within both case studies.

5.1 Introduction

Increasingly sophisticated analytic tools invade modern workplaces. These tools collect a large verity of data on employees, partners and clients to different purposes spanning from workforce intelligence (e.g., optimized recruiting, talent management, turnover impact assessment) to cybersecurity (e.g., next-generation firewalls, security information and event management systems). Although extremely useful, sometimes, vital to the organization, these tools often raise numerous ethical privacy issues [26] since the data collected often contain sensitive and personal information. Therefore access should be limited to the data relevant to the task at hand as mandated by data protection regulations. To this end, data need to be pre-processed to eliminate or obfuscate the sensitive information. Additional security/accountability measures may be also applied to reduce the privacy risk, such as logging the access to the personal data or imposing deletion obligations.

Anonymization is a commonly used practice to reduce privacy risk, obfuscating, in part or completely, the personal identifiable information in a dataset. Anonymization methods include [41]: suppressing part of or entire records; generalizing the data, i.e., recoding variables into broader classes (e.g., releasing only the first two digits of the zip code) or rounding/clustering numerical data; replacing identifiers with random values (e.g., replacing a real name with a randomly chosen one).

Anonymization increases protection, by lowering the privacy risk, and enables a wider exploitation of the data. However, anonymization techniques should be carefully applied since they clearly impact the quality (utility) of the data. Accordingly, different level of anonymization should be considered depending on a number of factors, often known at run-time only, such as the trustworthiness of the requester or security context of the query.

In this Chapter, we propose and demonstrate a privacy-aware risk-based access control framework, which addresses the concerns described above. This framework is an adaptation of the risk-based access control model proposed in the previous chapter, to the privacy context. In our framework, access-control decisions are based on the privacy risk level associated with a data access request. This risk level is dynamically evaluated using widely used privacy metrics (e.g., k-anonymity). and (if needed) anonymization is applied

on the specific resulting data set.

The inclusion of on-the-fly anonymization allows for extending access to the data, still preserving privacy below the maximum tolerable risk. Risk thresholds can be adapted to the trustworthiness of the requester role, so a single access control framework can support multiple data access use cases, ranging from sharing data among a restricted (highly trusted) group to public release (low trust value). Besides trust enhancement strategies can be applied to offer further flexibility in very delicate contexts (e.g., a security emergency, medical emergency) in return further guarantees should be supplied to ensure the leaked data will not be misused.

In addition to its flexibility, the proposed approach has a number of other advantages:

1. it provides a simple framework to address the, often conflicting, privacy and utility requirements;

2. it allows to easily set the risk and trust levels, and configure the adjustment strategy to meet the priorities of the organization (e.g., optimize utility or performance goals).

3. it can be easily integrated to widely used access control systems such as RBAC and ABAC.

To evaluate the feasibility and effectiveness of this approach we selected two cases studies namely "*HR information disclosure*" and "*Privacy aware threat investigation*". We developed two prototypes based on a slightly different version of the framework and run a set of experiments on each implementation.

In the first case study, we simulate the behavior of our framework in a scenario of an employee survey results dissemination. Then we assess the performance and impacts on the utility of a first version of the prototype by running a number of queries against the Adult DataSet from the UCI Machine Learning Repository, a publicly available dataset that is widely used by the research community. The experimental results were encouraging and confirm the feasibility of our proposed approach.

The second cases study (*Privacy aware threat investigation*) addresses the exploitation of employees network and system logs in the context of cybersecurity. In fact, due to the increasing complexity and variety of attacks, modern Threat Detection Systems (TDS) are becoming more sophisticated and data-intensive. They leverage the correlation of security events from several logs (collected from different sources in the organization's information system) to detect and prevent cyber attacks [123, 168]. This is typically done in two main steps: an automatic pattern or anomaly detection phase which highlights suspicious events followed by a detailed investigation carried out by a human expert who must decide whether the anomalous pattern corresponds to an actual attack. In this second phase, the expert must often inspect the raw data (log files) that triggered the alert.

Although the security investigation can constitute a legitimate purpose for their processing of the log data, whenever they contain sensitive or personal information (e.g., user ids, IP addresses, logins) access must be restricted on need-to-know bases. Therefore data must be anonymized to obfuscate those elements that are not strictly necessary for the task at hand. However, the application of anonymization techniques can deteriorate the quality or utility of the data. Although some analytics can still be run on anonymized log data [87], in many cases the anonymization affects the quality of results and, ultimately, decreases the ability to detect and react to cyber threats. By using our framework to control the data leakages, we do not require an *a priori* risk mitigation measure anymore, i.e., off-line, anonymization of the data sources. The automatic pattern detection phase uses the original data set, and anonymization is applied only if a subsequent, human-based analysis is needed on the resulting data.

In the next section (Section 5.2), we present the privacy-aware model. Section 5.3 we will describe the *HR information disclosure* case study and show how the framework can be combined with an RBAC access control model and applied to this case study. In Section 5.4 we report some preliminary results of the performance and impact on data quality of this model. In Section 5.5 we describe the second case study (i.e.,*Privacy aware threat investigation*) . In this second case study, our model is implemented on top of an attribute-based access control model. Section 5.6 discusses the results of an experimental evaluation of the proposed approach in terms of performance, scalability, and data utility (after anonymization). We describe in Section 5.7 how risk- based authorizations can be expressed through attribute-based policies and we provide some policy examples Lastly, we conclude in Section 5.8 with some final remarks.

5.2 Privacy-aware risk-based access control

In the privacy-aware risk-based access control we propose the access requests are assessed according to the authorization function we defined Eq. 4.1 the last chapter (Chapter 4).

$$Auth_{\Pi}(req) = \begin{cases} \textbf{deny} & \text{if } \Pi(req) = \mathsf{denied} \\ \sigma(req) & \text{if } T(req) - R(req) < 0 \\ \textbf{grant} & \text{otherwise} \end{cases}$$

We will, however, introduce and use a risk model R, a trust model T, and a set of trust and risk adjustment strategies σ specific to the context of data privacy. In the following sections, we will start by introducing privacy concepts that will be used in risk assessment and mitigation. Then we will refine the risk and trust model based on these concepts. finally, we will discuss some privacy preserving adjustment techniques.

5.2.1 Risk Model

As mentioned in the previous chapter, risk can be expressed in terms the likelihood of the occurrence of certain (negative) events system and the (negative) impact of these events [60]. In this chapter, we focus on the risk associated with privacy breaches in information systems. Privacy breaches are often associated with *individual identifiability*, used in most data protection privacy laws (e.g., the EU data protection directive [59], the Health Insurance Portability and Accountability Act (HIPAA) [141]). To prevent individual identifiability the regulation requires that disclosed information (alone or in combination with reasonably available information from other sources or auxiliary information [116]) guarantee a certain level of anonymity i.e., it should not allow an intruder to identify individuals in a dataset (identity disclosure) or to learn private/sensitive information about individuals (attribute disclosure) with a very high probability or confidence (see [148, 158]).

To quantify the level of anonymity of a respondent in a dataset, various syntactic anonymity metrics have been proposed in the literature (see [21, 43] for a review), the most popular being k-anonymity [134], ℓ-diversity [99], and t-closeness [93]. These metrics differ in a number of ways, but they all express the risk of disclosing personal-identifiable information when granting access to a given dataset.

Assuming a data represented as a relational table, called *private table*, where each record in the table is relative to a specific *respondent*, the above anonymity metrics propose to classify attributes (columns) in the table as follows:

- *Identifiers*. These are data attributes that can uniquely identify individuals. Examples of identifiers are the Social Security Number, the passport number, the complete name.
- *Quasi-identifiers (QIs) or key attributes* [47]. These are the attributes that, when combined, can be used to identify an individual. Examples of QIs are the postal code, age, job function, gender, etc.
- *Sensitive attributes*. These attributes contain intrinsically sensitive information about an individual (e.g., diseases, political or religious views, income) or business (e.g., salary figures, restricted financial data or sensitive survey answers).

The k-anonymity condition requires that *every* combination of QIs is shared by at least k records in the dataset. A large k value indicates that the dataset has a low identity privacy risk, because, at best, an attacker has a probability $1/k$ to re-identify a record (i.e., associate the sensitive attribute of a record to the identity of a respondent). Consider now a table with a group of k records sharing the same combination of quasi-identifiers have the same sensitive attribute. Even if the attacker is unable to re-identify the record,

he can discover the sensitive information (attribute disclosure). The ℓ-diversity metrics was introduced to capture this type of risk. It requires that for *every* combination of key attributes there should be at least ℓ values for each confidential attribute. Although the ℓ-diversity condition prevents the attacker from inferring exactly the sensitive attributes, he may still learn a considerable amount of probabilistic information: if the distribution of confidential attributes within a group sharing the same key attributes is very dissimilar from the distribution over the whole set, an attacker may increase his knowledge on sensitive attributes (*skewness attack*, see [93] for details). To overcome the problem, t-closeness estimates this risk by computing the distance between the distribution of confidential attributes within the group and in the entire dataset. These measures provide a quantitative assessment of the different risks associated to data release, and each of them (or a combination thereof) can be applied to estimate privacy risk depending on the use case at hand.

In this chapter, we will use k-anonymity as anonymity metrics to present our ideas, but it must be emphasized that the approach can easily be adapted to use alternative metrics (including ℓ-diversity and t-closeness).

In presence of identifiers the re-identification likelihood (L_{id}) is clearly maximum (i.e., $L_{id} = 1$), but even if identifiers are removed, the combination of QIs can lead to the identification of individuals and this implies a high risk. The k-anonymity condition requires that *every* combination of QIs is shared by at least k records in the dataset. A large k value indicates that the dataset has a low re-identification risk because an attacker has a probability $L_{id} = 1/k$ to re-identify a data entry (i.e., associate the sensitive attribute of a record to the identity of a User). Therefore, the (re-identification) risk related to a k-anonymous data-view v is:

$$R(v) = L_{id}(v) \times I(v)$$

where I is the impact associated with the identification of the respondents in the dataset. The severity of the impact is often evaluated in terms of monetary cost but it can also be assessed by assigning severity levels (e.g., spanning from minimal to critical). In this paper we will evaluate the impact in the interval $[0, 1]$, where 0 is *minimal* impact and 1 is *maximal* impacts. For the sake of simplicity, we will set the impact $I = 1$ and consider the risk $R(v) = L_{id}(v)$ this will allow us to normalize the risk and the trust values to $[0, 1]$ (as we discuss in the previous chapter).

5.2.2 Trust Model

In a general context we defined the trustworthiness as the confidence we have that a requester will not misuse the resource they are granted access to (see Chapter 4, Section 4.3 for more details). In our framework trust plays the role of a risk threshold: trusted users

are allowed to take large risks.[1]

We will use this definition as well in the context of data privacy, the risk threshold should be set to ensure a requester will not use the data leaked to violate the privacy of a respondent. Consequently, the trustworthiness of a request should reflect the set of parameters that will ensure a requester/an access context is not violating privacy.

Among these parameters we can name for requester ($T_{user}(u)$): requester behavior (with precedent accesses), requester seniority in a position and rank in a hierarchy[2], her experience and training in dealing with private data. The requester competence and tasks are also a very important parameter to take into consideration. Indeed Following the data minimization policy, the requester should have *enough* trust to access the resources (data) needed to the fulfill a set tasks she is expected to fulfill according to her role/competence. In the two use cases Sections 5.3 and 5.5 we will see some examples of how we assess T_{user} taking into consideration these parameters.

Regarding the request's context trust ($T_{context}(C)$): This trust value should (among others) reflect the level of safety of the environment in which the data will be released. This could depend on the device used to request access, the communication protocols etc. In addition, the urgency of the context the data is requested in (e.g., a cybersecurity emergency, medical emergency) can justify the need to access (or access more) data. Therefore in such contexts, the request trust is expected to be higher, and this is justified by a stronger need-to-know requirement.

For example the case study *"Privacy-aware threat investigation"* Section 5.5,we consider two main tasks namely *"Perform Maintenance and Improvement tasks"* and *"React to a Security Incident"*. The first task is expected to be regularly carried, and it is not linked to a security alert thus the access context is a *noAlert* context with a $T_{context}(noAlert)$). The task *"React to a Security Incident"*, instead,o be fulfilled in the context of *Alert* when a security threat of anomaly is found. This context has the trust level $T_{context}(Alert)$). It is easy to see that the two contexts have different access requirements. Indeed, in the latter, the need to react to a security threat overcomes the privacy requirements. The request could be granted access to sensitive data and therefore can be given a higher level of trust $T_{context}(Alert) \geq T_{context}(noAlert)$.

To compute the request trustworthiness (*total trust value*) we can use the approach for multi-dimensions trust computation proposed in [95], where the total trust is computed as a weighted sum of trust factor values.

$$T = \sum_{i=1}^{n} W_i \times T_i(\beta_i)$$

where β_i, $T_i()$, and W_i is a trust factor, a trust function and the weight of the i-th trust

[1] See [76] for a survey of different approaches to defining trust.

[2] See [45, 117] for a review on trust factors in the organizational context.

factor for $i = 1, \ldots, n$ respectively, subject to the constraint $\sum_{i=1}^{n} W_i = 1$. In our case, $n = 2$ and we can express our total trust value as:

$$T(q) = W \times T_{user}(u) + (1 - W) \times T_{context}(c)$$

5.2.3 Trust and Risk adjustment strategies

We presented the general idea of trust and risk adjustment in the previous chapter (Chapter 4 in Section 4.5) as well as some techniques used to decrease risk level and/or increase trust. In the following, we will provide more detailed about risk mitigation and trust enhancement that could be used in the context of data privacy. we will also discuss the effects of some of these techniques.

Risk mitigation. A possible way to decrease the privacy risk is anonymization (this enables to lower the likelihood of re-identification of respondents in a dataset). Anonymization can be achieved through obfuscating, in part or completely, the personal identifiable information in a dataset. Anonymization methods include [41]:

- *Suppression:* Removal of certain records or part of these records (columns, tuples, etc., such name/last name column);
- *Generalization:* Recoding data into broader classes (e.g., releasing only the first two digits of the zip code or replacing towns with country or regions) or by rounding/-clustering numerical data;

Besides anonymization other operation can be applied to the data, before its release, to reduce the privacy risk. Among these operations we can mention:

- Data perturbation: can be partial (e.g., for instance in IP addresses we can perturbation the machine address and preserve the network address) or total. It can be achieved through randomization or noise addition
- Pseudonymization: is a technique where identifying information are replaced by one or more artificial values or pseudonyms. Real information and pseudonyms can have a one-to-one mapping or in a more sophisticated way, the identifying information can have several pseudonyms to avoid linking several records to the same individual (same pseudonym). Pseudonymization can also be static meaning that the pseudonyms are set for the whole lifetime of a data set, or dynamic in which case pseudonyms are cyclically replaced after a period of time.

The risk mitigation operations (mentioned above) has an important computational overhead (besides other effects that we will discuss later) and therefore they are usually

run offline. However new technologies and products proposed more recently would facilitate the implementation of an on-the-fly, flexible privacy risk mitigation. Among these products, we can find the in-memory databases combined with column-store optimized algorithms provided by SAP Hana that can be easily integrated with new data-intensive business applications.

Trust enhancement. As mentioned in the previous chapter, trust enhancement will result in increasing the risk threshold and consequently exposing more data (more sensitive data). Therefore trust enhancement operations in the context of privacy need to ensure that the exposed data is processed in compliance the legal requirements. For instance, logging and monitoring access might be used to enhance "user trust" (T_{user}). Usage control operations controlling the retention period and ensuring the deletion of data can be used to enhance the "context trust" ($T_{context}$). The "context trust" can also be enhanced by informing respondents that their data will be logged and processed, specifying the access purpose to them, and if needed requesting their consent.

Adjustment strategies selection. It must be noted that adjustment strategies normally bring some *negative* side effects. For example, anonymization degrades data quality and this may affect its utility. Privilege escalation can increase the complexity of the security governance. Thus to identify the best adjustment strategy it is necessary to strike a balance between the advantages brought by the application of the strategy and the associated side effects.

For instance, if we focus on data access and privacy risk and limit the adjustment strategies to anonymization, then we can find an optimal anonymization strategy $\hat{\sigma}$, among all the possible anonymization strategies, that allows for data access reducing risk (so fulfilling Eq. 4.1) and, at the same time, maximizing the utility after application of the strategy. This can be expressed as a utility-privacy optimization problem:

$$\begin{aligned} \hat{\sigma} &= \arg\max_{\sigma} U(obj') \\ &\quad \text{s.t. } req' = \sigma(req) \text{ and } R(req') \leq T(req') \end{aligned}$$

where obj' denotes the resource in the request generated by the adjustment strategy, i.e. $req' = (u', a', obj', C')$.

In practice, the number of mitigation strategies is often very limited. The optimization problem is therefore reduced to testing a small set of anonymization strategies and estimating (either on the basis of numerical thresholds or expert assessment) if the utility of the result is sufficient for the business task under consideration. If this is not the case, trust enhancement mechanism can be triggered.

5.3 HR information disclosure

In this section we propose a privacy-aware risk-based access control designed on top the RBAC model, to manage privacy preserving HR information release in a corporate environment. First we will present an case study describing the specificity of this environment. Then we will describe the Privacy-aware Risk-based RBAC framework and show how our framework can address issues described in the case study.

5.3.1 Employee survey use case

Employee surveys are a widely used instrument for organizations to assess job satisfaction, quality of management, people motivation, etc. Considering the possible sensitivity of data, surveys should be anonymous, meaning that the organization and management should not be able to identify how a specific employee responded. Usually, the organization—say, a large company—conducting the survey outsources the data collection to a third-party. When processing the data, the third-party has access to individual-level information, whereas the same data is not accessible to the company. To protect the anonymity of the survey, the company can access the data under the condition that *(i)* identifiers are removed and *(ii)* the number of respondents is larger than a certain threshold (usually between 10 and 25). Different splits of data can be requested (e.g., per organization, per job profile, etc.), but data are accessible only if the query results contains a number of respondents that is larger than the fixed threshold. On top of that, additional access control rule can be enforced, e.g., a manager would only see data referring to his/her team or department (provided that conditions *(i)* and *(ii)* are also fulfilled); an employee would be allowed to see overall (company results) only. As an example, consider a question like "Do you respect your manager as a competent professional?" with a five points scale (1 to 5). A manager could see the response of his/her team if at least, say, 10 people answered to it. If the manager decides to refine the analysis asking for data related to the people in his/her team AND with a "developer" role, again the response should be made available only if at least 10 respondents with that role answered to the question.[3] Current systems typically do not provide any data if the number of respondents is below the defined thresholds (for the specific role). In other words, in order to avoid the risk of disclosing too much information, an overly conservative approach is taken and risky queries are not permitted altogether. Ideally, the access control system should be able to provide the largest possible amount of information (still preserving anonymity) for any query. In practice, in presence of queries that might cause anonymity issues (i.e., not enough respondents, or more generally, too small a result set), the system should be able

[3] In real surveys single records are actually never shown, but just percentages, in this example it would be something like 10% answered 1, 25% answered 2, etc. Since the number of respondents is known, in practice, for one question, this equivalent of getting the data with no identifiers.

to quantify the disclosure risk associated with the query and compare it with whatever risk level has been set as the acceptable threshold. If the threshold is exceeded, the system could apply, for example, a "generalization" operation (making the query less specific), thus increasing the cardinality of the result set and reducing the risk of disclosing the identity of respondents. Of course, applying such an operation would not yield the *exact* data set the user asked for, but this method would: 1) provide some relevant (i.e., as close as possible to the original query) information to the user, and 2) preserve anonymity according to some pre-defined disclosure-risk levels (possibly linked to the requestor trust or role). In the next sections,we discuss how to implement such a system using risk-based access control, and anonymization mitigation strategies.

5.3.2 Privacy-aware Risk-based RBAC model

In this section we show how we can integrate our Privacy-aware risk-based approach in a Role-Based Access Control (RBAC) model. This integration is inspired by the Risk-Aware Role-Based Access Control (R^2BAC) that has been introduced in [34, 35].

Risk-Aware Role-Based Access Control. The R^2BAC model consists of the following components:

- a set of users U;
- a set of permissions P, usually representing action-object pairs;
- a set of access requests Q, modeled as pairs of the form (u, p) for $u \in U$ and $p \in P$;
- a set of *risk mitigation methods* $\mathcal{M}$, i.e., actions that are required to be executed to mitigate risk;
- a function σ mapping permissions into *risk mitigation strategies*, i.e., lists of the form $[(l_0, M_0), (l_1, M_1), \ldots, (l_{n-1}, M_{n-1}), (l_n, M_n)]$, where $0 = l_0 < l_1 < \cdots < l_{n-1} < l_n \leq 1$ and $M_i \in \mathcal{M}$ for $i = 0, \ldots, n$;
- a set of *states* Σ, i.e., tuples of the form (U, P, σ, π) where π abstracts further specific features of the state; for instance, in the Risk-Aware Role-Based Access Control (R^2BAC) model [34], π comprises the set of roles $\mathcal{R}$, the user-role assignment relation $UA \subseteq U \times \mathcal{R}$, the role-permission assignment relation $PA \subseteq P \times \mathcal{R}$, the role hierarchy $\succeq \subseteq \mathcal{R} \times \mathcal{R}$.
- a *Risk function:* $risk : Q \times \Gamma \rightarrow [0..1]$ such that $risk(q, \gamma)$ denotes the risk associated to granting q in state γ;

- an *Authorization function* $Auth : Q \times \Gamma \to D \times 2^{\mathcal{M}}$ with $D = \{\mathsf{allow}, \mathsf{deny}\}$ such that if $q = (u, p)$ and $\sigma(p) = [(l_0, M_0), \ldots, (l_n, M_n)]$, and γ the current state, then

$$Auth(q, \gamma) = \begin{cases} (d_i, M_i) & \text{if } risk(q, \gamma) \in [l_i, l_{i+1}), i < n, \\ (d_n, M_n) & \text{otherwise} \end{cases}$$

  where $d_i \in D$. Intuitively, if the risk associated with access request (u, p) is l, then *Auth* returns an authorization decision and a set of risk mitigation methods corresponding to the interval containing l.

Privacy-aware Risk-based RBAC. In this paragraph we describe our Privacy-aware Risk-based RBAC framework, we highlight the main differences with chen and cramptons model [34], and show how our model can be mapped to the general risk-based model presented in Section 4.3. Let $\mathcal{V}$ be a set of database views (or virtual tables). If v is a view, then $|v|$ denotes the anonymity of v according to some given metrics (e.g. k-anonymity). The higher is the value of $|v|$, the smaller is the risk to disclose sensitive information by releasing v. Thus, for instance, we can define *the (privacy) risk* of disclosing p to be $1/|v|$ and *the (privacy) risk* of disclosing v to u in $\gamma = (U, \mathcal{V}, \sigma, \pi)$ to be

$$risk_\gamma(u, v) = \begin{cases} 1 & \text{if not } granted_\pi(u, v) \\ 1/|v| & \text{otherwise} \end{cases} \tag{5.1}$$

where $granted_\pi(u, v)$ holds if and only if u is granted access to v according to π. For instance, if π is an RBAC policy $(U, \mathcal{R}, P, UA, \mathcal{R}A, \succeq)$, then $granted_\pi(u, v)$ holds if and only if there exist $r, r' \in \mathcal{R}$ such that $(u, r) \in UA$, $r \succeq r'$, and $(p, r') \in PA$. Note that the $p \in P$ is the permission to perform an action a on a data-view v ($p = (a, v)$). In this chapter we are only interested in the action *read* from the view v: For simplicity sake, from here on, we will use v to represent both the data-view an the permission $(read, v)$.

When the risk associated to the disclosure of a certain view $v \in \mathcal{V}$ is greater than the maximal accepted risk t, we can use obligations for obfuscating or redacting the view and thus bring the risk below t. In this paper we consider k-anonymization functions $\phi_k : \mathcal{V} \to \mathcal{V}$ for $k \in \mathbb{N}$ as risk mitigation methods, but functions based on other metrics can be used as well. Clearly $|\phi_k(v)| \geq k$ for all $v \in \mathcal{V}$. We then consider risk mitigation strategies of the form $\sigma(v) = [(0, \iota), (t, \phi_{\lceil 1/t \rceil}(.))]$, where $\iota : \mathcal{V} \to \mathcal{V}$ is the identity function (i.e., such that $\iota(v) = v$ for all $v \in \mathcal{V}$) and the following authorization decision function:

$$Auth_\gamma(u, v) = \begin{cases} (\mathsf{allow}, \phi_{\lceil 1/t \rceil}(\cdot)) & \text{if } risk(u, v) \geq t \\ (\mathsf{allow}, \iota) & \text{if } risk(u, v) < t \end{cases} \tag{5.2}$$

Auth always grants access but yields an anonymized version of the requested view if the risk is greater that the maximal accepted risk t. In other words, if user u asks to access

v, then access to v is granted unconditionally if $risk(u, v) < t$, otherwise an anonymized version of v, say $\phi_{\lceil 1/t \rceil}(v)$, is computed and returned to u.

This authorization function is easy to map with the general risk-based access control authorization formula (presented in Chapter 4 in Eq. 4.1). The main difference is that in Eq. 5.2 we do have a *deny* outcome. As discussed in Section 4.3 in RBAC it is not possible to explicitly deny access through the access policy (unlike other models e.g., attribut based access control ABAC, ACLs). However access is denied if it is not explicitly granted by a policy. following this traditional approach we can add a third case to Eq. 5.2 where $Auth_\gamma(u, v) = (\mathsf{deny})$ if not $granted_\pi(u, v)$. In this case there is no need to compute the request risk if it is not granted by the policy. Which may offer a high level of protection but restricts access to a predefined set of users.

example 7. *To illustrate assume Alice asks for a view* v_1 *such that* $|v_1| = 4$ *and that* $\sigma(v_1) = [(0, \iota), (t, \phi_{\lceil 1/t \rceil}(.))]$ *with* $t = 0.1$*, i.e.* $\sigma(v_1) = [(0, \iota), (0.1, \phi_{10}(.))]$*. It is easy to see that* $risk(Alice, v_1) = 0.25$ *and that* $Auth((Alice, v_1), \sigma) = \phi_{10}(v_1)$*. Alice then asks for a view* v_2 *such that* $|v_2| = 20$ *and that* $\sigma(v_2) = \sigma(v_1) = [(0, \iota), (t, \phi_{\lceil 1/t \rceil}(.))]$ *with* $t = 0.1$*, i.e.* $\sigma(v_2) = [(0, \iota), (0.1, \phi_{10}(.))]$*. It is easy to see that now* $risk(Alice, v_2) = 0.05$ *and therefore that* $Auth((Alice, v_2), \sigma) = \iota(v_1) = v_1$*.*

The following results state that the risk of disclosing the view returned by our authorization decision function is never greater than the maximum accepted risk.

In many situations of practical interest, we want the risk threshold t to depend on the trustworthiness $trust(u, v)$ of the query $q = (u, v)$. With $trust : U \times \mathcal{R} \times P \to (0..1]$ is a function that assigns a trust value to users. In the context of RBAC, roles correspond to job functions, it is natural to assign trust to roles and to derive the trust of a user from the trust assigned to the roles assigned to that user in the following way:

$$trust(u) = \max\{trust(r') : (p, r') \in PA \text{ and } \exists r \succeq r' \text{ s.t. } (u, r) \in UA\}.$$

5.3.3 Application of the model

We now show how our privacy-aware risk-based RBAC model can be used to support the case study of Section 5.3.1. This will be done by setting appropriate values to the parameters occurring in the definition of the risk function (5.1).

For sake of simplicity we consider a small company, with 8 employees and one manager. The company runs an employee survey, with one single question with answer ranging in a five points scale (from 1 to 5) (*sensitive attribute*, cf. Section 5.2.1), and collecting user names[4] (the *identifiers*), as well as the job title and the location of the office (the *quasi-identifiers*). The actual dataset is in Table 5.1(a).

[4]In real cases they are typically user IDs

Table 5.1: The Employee Survey Example

(a) Original dataset

Survey Administrator view $\lvert v_{all}\rvert = 1$			
Name	Job	Location	**Answer**
Timothy	SeniorDeveloper	Houston	4
Alice	Support	Houston	5
Perry	JuniorDeveloper	Rome	5
Tom	Admin	Rome	3
Ron	SeniorDeveloper	London	4
Omer	JuniorDeveloper	London	4
Bob	Support	Houston	5
Amber	Admin	Houston	3

(b) Anonymized version: *identifiers* and *quasi-identifiers* are suppressed

Employee View $\lvert v_{supp}\rvert = 8$			
Name	Job	Location	**Answer**
***	***	***	4
***	***	***	5
***	***	***	5
***	***	***	3
***	***	***	4
***	***	***	4
***	***	***	5
***	***	***	3

The outsourcing company collecting the data is considered fully trusted and will therefore have access to all the information. We model this by setting the trust of the `admin` role to 1, i.e. $trust(\texttt{admin}) = 1$. Thus, an administrator can access the original dataset, say v_{all} with anonymity $|v_{all}| = 1$ (i.e., all distinct values, see Table 5.1(a)). If we set the trust value of the `manager` role to 0.35, i.e. $trust(\texttt{manager}) = 0.35$ (corresponding to access views with anonymity $k \geq 3$), than a manager cannot access v_{all} as is, since $1 > 0.35$ and some anonymization, as risk mitigation strategy, must be carried out on the data to decrease the risk. For example, if we suppress the identifier attribute (*Name*) and the quasi-identifiers (*Job* and *Location*), we obtain the view v_{supp} shown in Table 5.1(b). The view v_{supp} corresponds to an anonymity level $|v_{supp}| = 8$ and since $0.125 < 0.35$, access is granted to the manager.[5] The manager can also ask for more granular views of the

Table 5.2: Views of the employee survey for the Rome location

(a) Before generalization.

View: Location=Rome, $\lvert v_{Rome}\rvert = 2$			
Name	Job	Location	**Answer**
***	***	Rome	5
***	***	Rome	3

(b) After generalization

View: Location=Rome Anoymized $\lvert v_{EMEA}\rvert = 4$			
Name	Job	Location	**Answer**
***	***	EMEA	5
***	***	EMEA	3
***	***	EMEA	4
***	***	EMEA	4

results. For example, if she wants to know the distribution of the answers in one location, say Houston, $|v_{Houst}| = 4$, the risk $0.25 < 0.35$ is still smaller than the *trust*. On the other

[5] In real surveys the result will appear as a report like: 37.5% answered 5, 37.5% answered 4 and 25% answered 3. For a single question this is equivalent to the view in Table 5.1(b).

hand, if she asks for the result in Rome, $|v_{Rome}| = 2$, then the risk associated with the view for the manager is $0.5 > 0.35$ and the access is granted only if appropriate anonymization is performed. In this case, location could be generalized from Rome to EMEA (so including London workforce), as shown in Table 5.2(b). The resulting view has anonimity $|v_{EMEA}| = 4$ and since the risk 0.25 is smaller than the *trust* ($trust(\texttt{manager}) = 0.35$), then the manager is allowed to see the view.

Table 5.3: Views of the employee survey for Rome and JuniorDeveloper

(a) Before generalization of location and job

Loc=Rome AND Job=JuniorDeveloper $\lvert v_{Rome+JuniorDeveloper}\rvert = 1$			
Name	Job	Location	**Answer**
***	JuniorDeveloper	Rome	5

(b) After generalization of location and job

View Loc=Rome AND Job=JuniorDeveloper Anonymized $\lvert v_{EMEA+Dev}\rvert = 3$			
Name	Job	Location	**Answer**
***	Dev	EMEA	5
***	Dev	EMEA	4
***	Dev	EMEA	4

Similarly, if the manager wants to see the results per location and per job function (say in Rome for JuniorDeveloper only, see Table 5.3(a)), the anonymity level is low, $|v_{Rome+JuniorDeveloper}| = 1$, and the associated risk is greater than 0.35. Again, instead of simply denying access, the system can perform generalization on both the quasi-identifiers, *Job* (generalized to the job family developer) and *Location*, thereby increasing the anonymity ($|v_{EMEA+Dev}| = 3$) and decreasing the risk ($risk(manager, v_{EMEA+Dev}) = 0.33$) to an acceptable level for a manager (see Table 5.3(b)).

Finally, employees should have access to the global results only. The trust value is therefore set to $trust(\texttt{employee}) = 0.125$ and the only view permitted is with suppression of all identifiers and quasi-identifiers, which has $|v_{supp}| = 8$, see Table 5.3(b).

5.4 Feasibility evaluation using the employee survey use case

This section documents the results of an initial evaluation of our approach. The two questions we investigate are (A) whether the approach described in this paper can be realized in practice and (B) whether the performance that can be expected under typical workloads matches the needs of real-time (more precisely: online) operation.

In order to address question *A*, we realized a prototype system that we have used to

run sample scenarios. We use the same prototype also to study the response time under several representative conditions (queries of varying complexity, different levels of user trust and therefore, different loads for the anonymizer module).

In the following, we first describe our prototype implementation, then we present the dataset we used for the evaluation and outline the results of the experiments we run on that dataset.

5.4.1 Prototype Implementation

In order to evaluate the practical feasibility of our approach, we developed a proof-of-concept implementation of our framework that we used to run the experiments described in the following.

Our prototype is implemented in Java 7 and uses MySQL Server version 5.6.20 to store the dataset. The Risk Aware Access Control module mimics a typical XACML data flow, providing a basic implementation of the PDP, the PEP, and the PIP functionality as well as a set of authorization policies. The Risk Mitigation module is implemented using the ARX[6] anonymization framework [85]. The ARX toolkit offers a Java API supporting data de-identification. ARX is capable of altering input data in a way that guarantees minimal information loss while ensuring that the transformed data adheres to well-defined privacy criteria, expressed in such metrics as k-anonymity, ℓ-diversity, t-closeness, etc. ARX also offers several reporting features allowing to collect metrics such as execution time, information loss, etc. We evaluated other available anonymization libraries (e.g., Cornell Anonymization Toolkit[7], University of Texas Anonymisation Toolbox[8]). We eventually adopted ARX because we found it easy to integrate and considering that it is a well-documented, actively developed, and well maintained project.

5.4.2 Dataset

To test the performance of our framework, we used a dataset that is widely used in the research community, namely the Adult Data Set [9] from the UCI Machine Learning Repository. This dataset contains 32561 records from the US Census dataset with 15 demographic and employment-related variables. We removed records with missing values, ending with $30,162$ usable records, and we reduced the number of fields to nine, as shown in Table 5.4.

The choice of the identifiers, QIs and sensitive attribute set, typically, depends on the specific domain. QIs should include the attributes a possible attacker is likely to have

[6] http://arx.deidentifier.org/overview/
[7] http://anony-toolkit.sourceforge.net/
[8] http://cs.utdallas.edu/dspl/cgi-bin/toolbox/index.php
[9] Available at http://archive.ics.uci.edu/ml/datasets/Adult

access to (e.g., using a phonebook or a census database), whereas sensitive attributes depend on the application the anonymized data are used for.

Generally speaking increasing the number of QIs increases the risk, or results in strong anonymization impacting the usefulness of the resulting view. In our experiments we set $QI \equiv \{\text{AGE}, \text{NATIVE-COUNTRY}\}$. In the census data, the SALARY-CLASS attribute is typically chosen as a sensitive attribute. We also classified RACE as a sensitive attribute because of its discriminatory nature.

Table 5.4: Summary of the dataset columns, number of distinct values, and nature of each column

UCI Adult Dataset		
Attribute	Values	Nature
AGE	72	QI
NATIVE-COUNTRY	41	QI
EDUCATION	16	not Sensitive
OCCUPATION	14	not Sensitive
WORKCLASS	7	not Sensitive
MARITAL-STATUS	7	not Sensitive
GENDER	2	not Sensitive
RACE	5	Sensitive
SALARY-CLASS	2	Sensitive

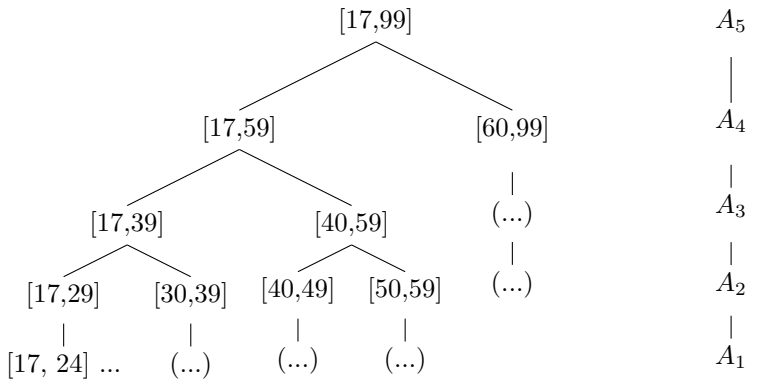


Figure 5.1: Generalization hierarchy for the attribute AGE [17, 99]. Level A_1: Age is generalized in 5 year range. Level A_2 in 10 year range. Level A_3 in 20 years. Level A_4 in 40 year range. In level A_5 the age is fully generalized. Age is not generalized in level A_0 (not shown).

QIs will be generalized according to the generalisation scheme of Figure 5.1 (for the attribute AGE) and Figure 5.2 (for the attribute NATIVE-COUNTRY).

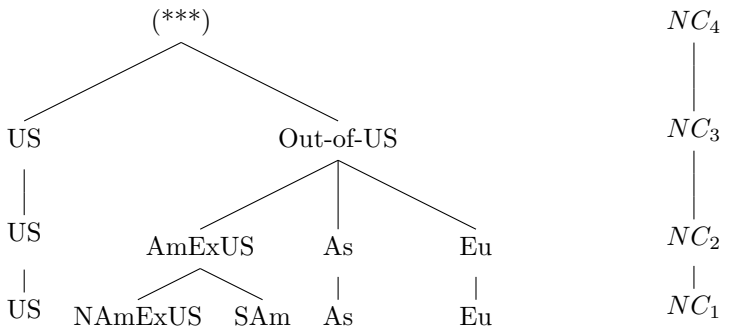


Figure 5.2: Generalization hierarchy for the attribute NATIVE-COUNTRY: Level NC_1: NATIVE-COUNTRY is generalized to US (United States), AmExUS (America Excluding United States), Asia (As), or Europe (Eu). Level NC_2: NAmExUS (North America Excluding United States) and SAm are generalized to AmExUS (America Excluding United States). Level NC_3: All countries excluding United States are generalized to Out-of-US. Level NC_4: native countries are suppressed. Level NC_0: native countries are not generalized (not shown).

5.4.3 Experiment and Results

In order to evaluate the performance of our tool, including the computational overhead caused by the anonymization engine, we used a number of queries of increasing complexity in terms of the size of the returned views and the disclosure risk. The queries are given in Table 5.5 and the corresponding size and anonymity level of the views returned by our tool are reported in Table 5.6. In the following we will indicate both the queries and the corresponding views as **Q1**, **Q2**, **Q3**, **Q4**.

For our experiments, we want to investigate the impact of risk mitigation, anonymization, on *(i)* the performance of the access control system and *(ii)* the quality of the resulting data. For case *(i)* we focus on the views with the largest sizes (namely, **Q1** and **Q2**, with more than 20,000 tuples each as shown in Table 5.6). For case *(ii)* we focus on the views with the highest risk profiles (namely, **Q1**, **Q3**, and **Q4**, with the lowest possible anonymity), whose computation is significantly affected by anonymization. We consider five risk thresholds *trust* i.e. users/role with different trustworthiness level, as shown in Table 5.7, and each experiment is run 100 times to average out the variance of the response time. In Figure 5.3 we report the results of the experiments for the four queries, panels **Q1**, **Q2**, **Q3**, and **Q4**, respectively, for the five different trustworthiness levels. Figure 5.4 shows the (possible) impact of generalization on the data accuracy, as measured by the Precision metric (Prec) [147], which counts the average number of generalization steps performed on the generalization trees (cf. Figure 5.1 and Figure 5.2).

For **Q1**, we observe that the anonymization process increases significantly the response time. Indeed the query is carried our by the most trusted user ($trust = 1$), with no anonymization needed, takes on average 8ms (see Figure 5.3.Q1, horizontally striped bar

Table 5.5: Queries

Q1: Data about male respondents

```
SELECT * FROM ADULT
WHERE SEX = ‘‘Male’’;
```

Q2: Data about adults between 30 and 75 years old born in the United States

```
SELECT * FROM ADULT
WHERE AGE BETWEEN 30 AND 75
AND NATIVE-COUNTRY = ‘‘United-States’’;
```

Q3: Data about adults between 30 and 35 years old working in the private sector and originally from the american continent excluding United States

```
SELECT * FROM ADULT
WHERE WORKCLASS = ‘‘Private’’
AND AGE BETWEEN 30 AND 35
AND NATIVE-COUNTRY IN
(<America Excluding the United-States>);
```

Q4: Data about adults without-pay

```
SELECT * FROM ADULT
WHERE WORKCLASS = ‘‘Without-pay’’
```

Table 5.6: Size and disclosure risk level of the views returned in response to the queries

Query	Size	Anonimity	Risk level
Q1	20,380	1	High
Q2	19,392	32	Low
Q3	215	1	High
Q4	14	1	High

Table 5.7: User roles and trustworthiness

User Name	Role	Trustworthiness
Alice	SuperUser	1
Megha	Admin	0.52
Dana	SeniorDataAnalyst	0.1
Frida	JuniorDataAnalyst	0.028
Eliyes	IT	0.015

corresponding to $trust = 1$). By decreasing the trustworthiness of the requester the view must be anonymized and the average response time increases to 27ms (cf. Figure 5.3.Q1, horizontally striped bar corresponding to $trust = 0.52$). This time difference is entirely due to the anonymization time (19 ms, as shown in Figure 5.3.Q1, diagonally striped bar

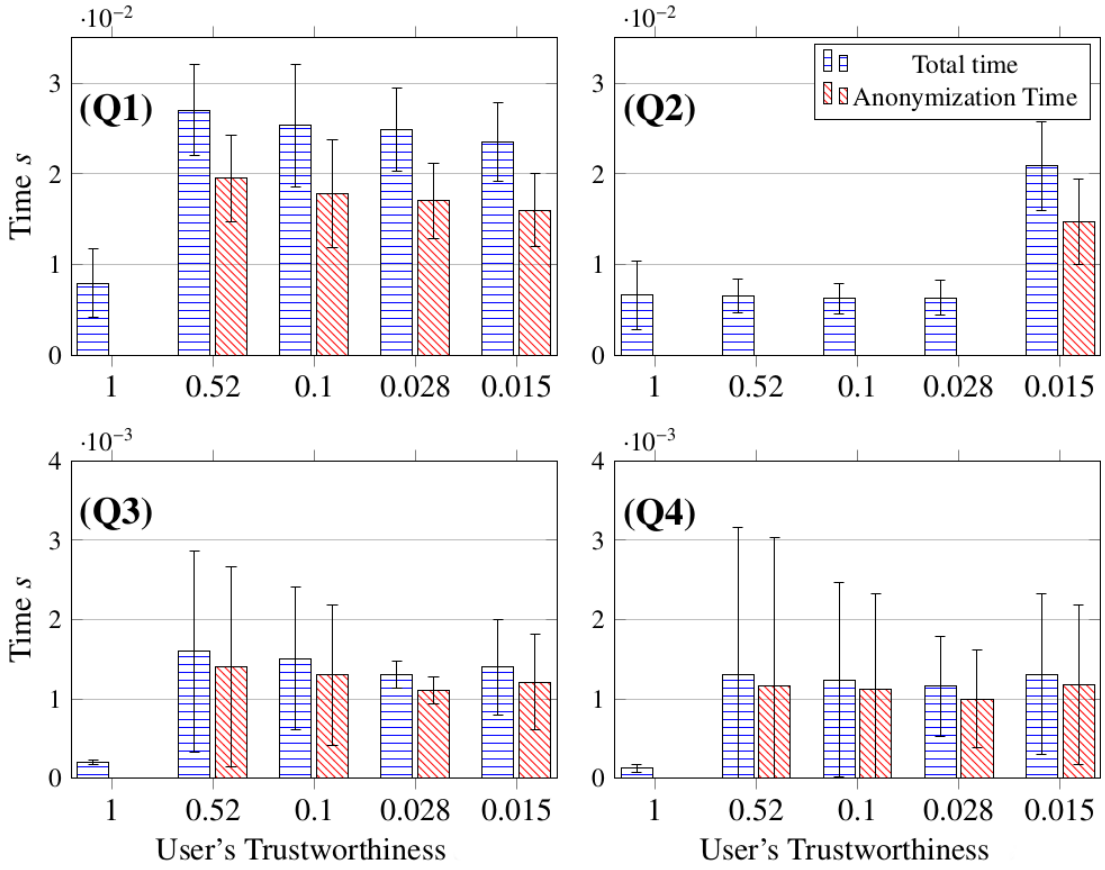


Figure 5.3: Average total response time (horizontal striped bars) and average anonymisation time (diagonally striped bars) for the four views and different trust levels.

corresponding to $trust = 0.52$). Decreasing further the trust level results in additional anonymization. Also the attribute NATIVE-COUNTRY (NC) gets anonymized (cf. Figure 5.4.Q1), but this does not significantly affect the response time (see Figure 5.3.Q1).

We can observe a similar behavior in the other queries (see Figure 5.3.Q2, Q3, and Q4), with an increase of response time when anoymization takes place and no significant variations in performance for different levels of anonymization. For instance, for **Q2** we have a view with an already high level of anonymity ($k = 32$), and a *small* anonymization (a single level of generalization for the Age attribute, see Figure 5.4.Q2 for $trust = 0.015$) still significantly impacts the performance. In case of **Q3** we see that, despite different combinations of anonymization strategies for different values of *trust* (Figure 5.4.Q3), the response time is not affected (Figure 5.3.Q3), except for $trust = 1$ where we have no anonymization. We should note that for **Q3** (as well as **Q4**) the difference in the average response time with and without anonymization is relevant ($trust = 1$ has response time of 0.16ms, and $trust = 0.52$ of 1.6 ms) but these views have few tuples and these times are

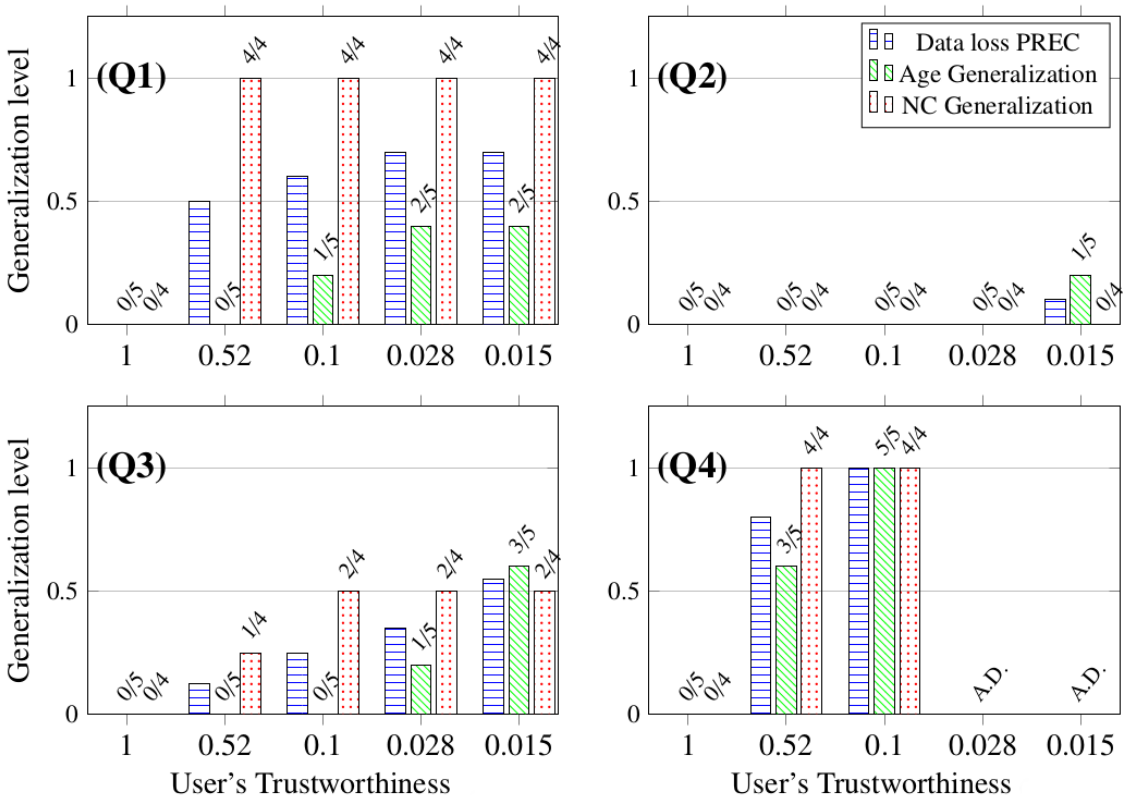


Figure 5.4: Generalization levels for the four views. Horizontal striped bar shows PREC metric (see text), diagonally striped bar the level of generalization for Age attribute and dotted bar the level of generalization for Native Country attribute. A.D. stands for Access Denied.

small in absolute value, with large fluctuations, as shown by the high standard deviations.

Q4 is characterized by a low cardinality and (consequently) by high anonymity. Except for the maximum trust value the data are strongly anonymized and for low trust levels $trust = 0.28$ and $trust = 0.015$ access is denied in spite of the anonymization, see Figure 5.4.Q4. Note that in these cases, the anonymization engine tries to minimize the risk (anonymization time is not zero, see Figure 5.3.Q4), but due to the low cardinality no solution is found.

From these experiments, we observe that when anonymization is applied the response time increases significantly, but, even in the worst cases, the increase is far less than one order of magnitude with no impact on the real-time response of the system. Moreover, the application of different anonymization strategies have no impact on the response time.

The experiments were carried out using a MacBook Air, operating system OS X 10.8.5, processor 1.3GHz Intel Core i5, memory 8GB 1600Mhz DDR3 and flash storage 120GB.

5.5 Privacy aware threat investigation

Modern intrusion detection systems at application-level (called Threat Detection System, TDS, herein)[10], collect security information on the application stack and correlate it with context information to detect potential threats. Usually, a TDS first collects application-level log files from various sources, enriches the data gathered from logs with contextual information (e.g., time and location), and finally stores the resulting data in a database. The events data are then automatically analyzed on a periodic basis against pre-defined threat patterns to detect potential anomalies and attacks. A pattern represents a combination of suspicious log events that could indicate a threat. Often it is defined as a set of filters applied to the event database and compared with some thresholds. If the threshold is exceeded, then an alert is triggered. For instance, the ensemble of events indicating a *Failed Login* initiated by the same source (e.g., Terminal) may indicate that a *Brute Force Attack* is underway if the number of attempts exceeds, say, 20 attempts in less than 10 minutes. When an alert is raised a human operator is asked to step in in order to evaluate if the alert corresponds to an actual threat and when this is the case to undertake appropriate countermeasures. To carry out his task, the operator may require access to the details of the data that triggered the alert. The operator should be granted access to sensitive data if this is strictly necessary to carry out her task and the severity of the problem justifies it.

Figure 5.5 illustrates the architecture of the system as well as the different users involved in the process. In Table 5.8 we provide an example of user/roles interacting with the TDS and the corresponding access authorizations required to execute their tasks.

Table 5.8: Roles

Operator	Classify alerts and report patterns anomalies His/Her tasks require access to pattern detection results (events/log data related to the suspicious pattern) in case of alerts.
Administrator	Has all *Operator* tasks and privileges. They can also Investigate alerts, Create or Reconfigure patterns. He/She should have access the detection results and events data related to the patterns.
Advanced Administrator	Has all *Administrator* tasks and privileges. Can also grant exceptional access to the data by attributing higher trust level to an *Operator* or an *Administrator*.

Although log files may contain personal information (e.g. names, IP addresses) investigation can constitute a legitimate purpose for their processing. Yet access to sensitive

[10]We refer to these systems as TDS, to distinguish them from network-level intrusion detection systems (often called IDS or SIEM). We base our description on the SAP Enterprise Threat Detection, but the analysis can be applied to other solutions, including IDS. For a comparison between application- and network-level intrusion detection systems, see [79].

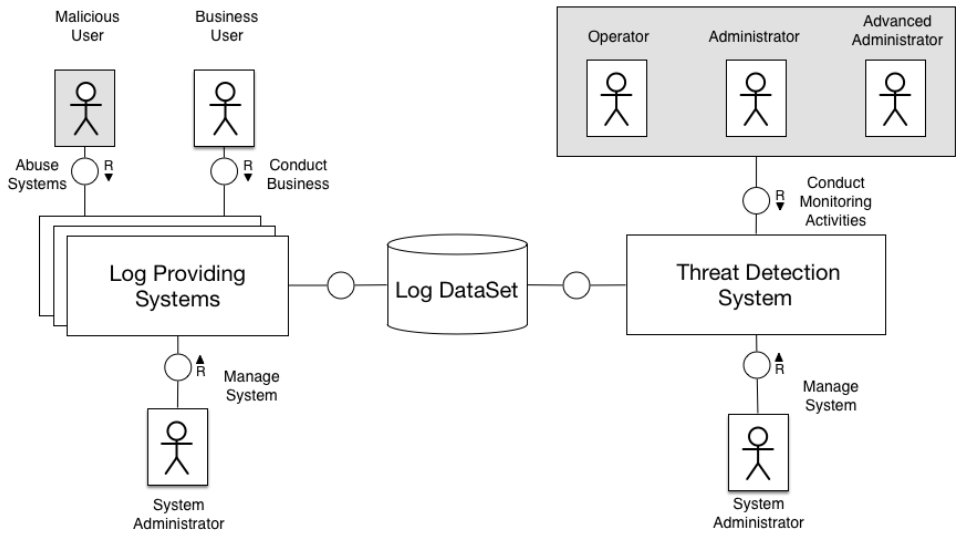


Figure 5.5: Business Roles and System Landscape

data should be done according to the data minimization principle, i.e. that access to personal information should be limited to what is directly relevant and necessary to accomplish the specified purpose. This is usually achieved in TDS by carrying out some (pseudo-)anonymization before analyzing the event data, such as replacing real user name or IDs with pseudonyms.

Still, with the increasing variety and complexity of collected log files, a full anonymization of the log dataset before processing could, on one hand, provide a good privacy protection, but also significantly impact the performance of the system, both in terms of the *utility* (the quality of results of the pattern detection phase, or the information available to the operator for the manual inspection) and processing time (anonymization on large data set could be time-consuming, and on data stream re-run regularly)

To address this challenge, a more dynamic approach is needed: instead of anonymizing the complete event data-base beforehand, whenever a user performs an operation accessing event tables, we have to apply specific anonymization methods which reduce the privacy risk but preserving the most relevant information for that operation. In practice, the anonymization process should be customized for each operation (to preserve the information useful for completing the task) and for each type of users, which can have different levels of access to the data. In the next section, we will propose a framework that to realize this scenario.

5.6 Experimental Evaluation

In this section wi will use the second case study described in Section 5.5 to assess the performance of our approach (described in Section 5.2). The TDS is expected to provide accurate real-time results, therefore we investigate the impact of our approach on the functioning of the TDS, in particular, whether the expected *Performance* and *Utility* matches the accuracy and real-time requirements.

More in details, as mentioned in Section 5.5, the TDS allows to detect potential attack patterns automatically, and then if additional investigations are needed, a human operator can browse the log data of the events corresponding to a given pattern for manual inspection.

Ideally, the operator should be able to perform the manual investigation (i.e., decide if the detected threat is a false or true positive). Some investigations can be conducted on data where the personal information are anonymized (or in any case, where the re-identification risk is low). If the operator does not have sufficient information to decide, he/she should be granted access to less anonymized (riskier) data, or in other words get higher access privileges (trust enhancement) acquiring administrator rights, or directly involving an administrator.

Accordingly, we need to check:

- *Utility*. Does the model allow a low trusted operator (i.e., small risk threshold) to perform the investigation in most cases, and relying on trust enhancement for the remaining cases?

- *Performance*. Does the additional anonymization step impact real-time performance?

Before addressing these questions (see Section 5.6.6), we need to describe our prototype implementation (Section 5.6.1), the data set and its attributes classification from a privacy risk perspective (Section 5.6.2), the selection of typical patterns used for the validation (Sect. 5.6.3), the utility measure (Sect. 5.6.5) and the trust level setting (Section 5.6.4).

5.6.1 Prototype Implementation

Our prototype is implemented in Java 8 and uses SAP HANA Database. It is composed of 3 main modules:

- The *Risk Aware Access Control module:* mimics a typical XACML data flow, providing an implementation of the PDP, the PEP, and the PIP functionality as well as a set of authorization policies.

- The *Risk Estimation module:* evaluates the privacy risk using pre-configured criteria (privacy metrics, anonymization technique, identifying information). It compares

Table 5.9: An extract of the Log dataset columns, privacy classification of each column and anonymization technique to be applied

Log Events data set		
Attribute	Type	Anonymization
EventID	Non-Sensitive	
Timestamp	Sensitive	
UserId (Origin)	Identifier	Suppression
UserId (Target)	Identifier	Suppression
SystemId (Origin)	QI	Generalization
SystemId (Target)	QI	Generalization
Hostname (Origin)	QI	Generalization
IPAddress (Origin)	QI	Truncation
MACAddress (Origin)	QI	Truncation
TransactionName	Sensitive	
TargetResource	Sensitive	

the privacy risk to the request trustworthiness level, then produces an estimation of the minimal anonymization to be applied in order to meet this level.

- The *Trust & Risk Adjustment module:* we implemented the Risk Adjustment Component to perform anonymization. It uses ARX [86] a Java anonymization framework implementing well-established privacy anonymization algorithms and privacy metrics such as k-anonymity, ℓ-diversity, t-closeness, etc. (the Trust Adjustment Component was not implemented in this version of the prototype.)

5.6.2 Data Set and privacy classification

To test the performance of our framework in the TDS use case, we used a dataset containing around $1bn$ record of log data collected from SAP systems deployed in a test environment [11]. The logs dataset is composed of 20 fields (in Table 5.9 we present a summary of the most important fields)

As described in Section 5.2.1, to assess the privacy risk of releasing a dataset, we first need to formalize our assumptions on the attributes that can be used to re-identify the entry, or, in other words, classify the attributes in terms of identifiers, QIs, and sensitive attributes. This classification, typically, depends on the specific domain. QIs should include the attributes a possible attacker is likely to have access to from other sources, whereas sensitive attributes depend on the application the anonymized data are used for. For example, in our experiments, we set (obviously) User ID as an identifier and the IP address as a quasi-identifier. Similarly, we assume that the Transaction name (the called function) cannot provide any help for re-identification, therefore we consider it a sensitive

[11] For an analysis of the performance of the model on a benchmark dataset see 5.4.

Table 5.10: Queries: Resulting views Size and Risk level

Query	***Corresponding Pattern***	***View Size***	***Risk Level***
Q1	Brute Force Attack	Large (50550)	Very High ($k = 2$)
Q2	Security Configuration Changed	Large (40300)	Medium ($k = 7$)
Q3	Blacklisted Function Called	Medium (14500)	Very High ($k = 1$)
Q4	Table Dropped or Altered	Small (228)	Medium ($k = 6$)
Q5	User Assigned to Admin Group	Very Small (12)	Very High ($k = 1$)

attribute (and no anonymization will be applied). Table 5.9 provides an example of this classification, and, for identifiers and quasi-identifiers, the corresponding anonymization methods applied.

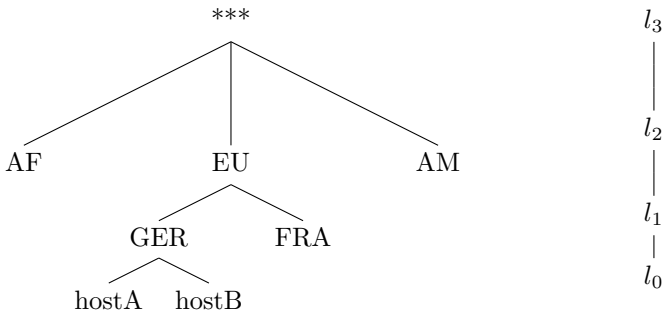


Figure 5.6: The generalization hierarchy for host names is organized as following: l_1 and l_2 are a location based generalization by country then by continent. in level l_3 host names are totally obfuscated and entirely revealed at the level l_0.

5.6.3 Pattern detection and investigation

In our experiments, we focus on 5 typical *Patterns* with different complexity in terms of the size of the returned data-views and the privacy risk. Two different kinds of queries are used during each phase respectively *Detection Queries* and *Investigation Queries*. The selected queries {**Q1** ... **Q5**} described in Table 5.10 are all *Investigation Queries*. An *Investigation Query* is a "SELECT *" extracting all the details of the events corresponding to certain pattern.

5.6.4 Roles and Trustworthiness levels

We have 3 roles Operator, Administrator and an Advanced administrator with increasing access requirements (to fulfill their tasks), therefore increasing privacy clearances, (i.e., larger risk tolerance). Usually, for k-anonymity, k values in the range $3-10$ are considered medium risk, $k > 10$ low risk, and for $k \leq 2$ the risk is very high (clearly, for $k = 1$ the

Table 5.11: Users/Roles Privacy clearances and Trustworthiness levels

Role	***Access Requirement***	***Privacy Clearance***	***Trust Level*** (Risk Threshold)
Operator	Low	Minimal ($k > 10$)	$T_u \in [0.05, 0.1[$
Administrator	Medium	Medium ($k > 2$)	$T_u \in [0.1, 0.5[$
Advanced administrator	High	Maximum ($k \leq 2$)	$T_u \in [0.5, 1]$

risk is the maximum, no anonymity) [122] . Therefore we propose the parameter setting described in Table 5.11, where for sake of simplicity we have considered a single trust factor $T = T_u$ (i.e. we set $W = 1$ in Eq. 5.2.2).

5.6.5 Utility Evaluation

The effect of anonymization in terms of utility is a widely discussed issue in the literature several generic metrics have been proposed to quantify the *"damage"* caused by anonymization (see [63] for a review). However, these metrics do not make any assumption on the usage of the data (so-called *syntactic metrics*), limiting their applicability on realistic use-cases.

Other approaches propose to assess the accuracy loss (Utility loss) of a system (i.e., IDS in [88], Classifier in [24]) by comparing the results of certain operations run on original then anonymized dataset using use case related criteria (i.e., in the context of a TDS the comparison criteria can be the number of *False positives*). Although interesting for our context, this approach cannot be applied in our use case, since it assumes that the analysis is run directly on anonymized data, whereas, in our use case, the pattern detection is performed on *clear* data, and the anonymization is applied only on the results (data-view).

We propose a method combining both approaches and that would include an evaluation:

- *From Syntactic standpoint:* The information loss caused by the anonymization, we use the precision metric that allows us to estimate the precision degradation of QIs based on the level of generalization with respect to the generalization tree depth (e.g., for the generalization tree 5.6 if we allow access to continent instead of host-names we used the 3^{rd} level generalization out of 4 possible levels so $d_p(hostnames) = 3/4 = 75\%$ precision degradation for host-names).

- *From Functional standpoint:* The effect of this loss on our use case. During the investigation phase, the operator, mostly, bases their analysis on a subset of attributes, which are different for each attack pattern. Thus we will assign a utility coefficient uc to different attributes based on the relevance of the attribute to the pattern/query.

Combining the two approaches we compute the utility degradation of a data-view v as

$$U_d(v) = \sum_{a_i \in A} uc_{a_i} \times d_p(a_i) \tag{5.3}$$

with $A = \{a_1..a_i\}$ the set of attributes in the data set. We also set the precision degradation of the identifiers to $d_p(identifiers) = 1$ as they will be totally suppressed after the anonymization.

5.6.6 Results and Analysis

For our experiments, we want to investigate: *(i)* Performance: the impact of on-the-fly anonymization (as risk mitigation strategy) on the performance (response time). *(ii)* Utility: we would like to investigate if the quality of resulting data is generally enough to fulfill the expected tasks for every user/role for various pattern investigation.

In order to evaluate these aspects we run several experiments considering 5 patterns and 7 users/role with different trustworthiness level, $t = \{0.055, 0.083\}$ Operators, $t = \{0.12, 0.15, 0.45\}$ Administrators, and $t = \{0.9, 1\}$ Advanced Administrators. The corresponding size and anonymity level of the views returned by the queries (corresponding to the selected patterns) are reported in Table 5.10. In the rest of this section we will indicate both the queries and the corresponding views as **Q1**, **Q2**, **Q3**, **Q4** and **Q5**.

Performance and scalability To evaluate the performance of our tool, including the computational overhead caused by the anonymization, we run queries **Q1**, **Q2**, **Q4**, and **Q5** (described in Table 5.10) using our access control prototype experiment, 100 times for each query to average out the variance of the response time. In Figure 5.7 we report the results of the experiments for the four queries for the 6 trustworthiness levels.

For **Q1**, we observe that the anonymization process increases significantly the response time. In fact when the query is carried out by the most trusted user ($t = 0.9$), with no anonymization needed, the response time on average is less than 15ms (see Figure 5.7.Q1, the diagonally striped bar corresponding to $t = 0.9$). By decreasing the trustworthiness of the requester the view must be anonymized and the average response time increases to 150ms in the worst case (cf. Figure 5.7.Q1, the diagonally striped bar corresponding to $t = 0.055$). This time difference is entirely due to the anonymization time (130 ms, as shown in Figure 5.7, **Q1**, horizontal striped bars corresponding to $t = 0.055$). Increasing the trust level decreases the needed anonymization, but it slightly affects anonymization time. We can observe a similar behavior in the other queries (see Figure 5.7, **Q2**, **Q4**, and **Q5**), with an increase of response time when anonymization takes place and no significant variations in performance for different levels of anonymization. For instance, for **Q2** and **Q4** we have two views with an already medium level of anonymity (respectively $k = 7$

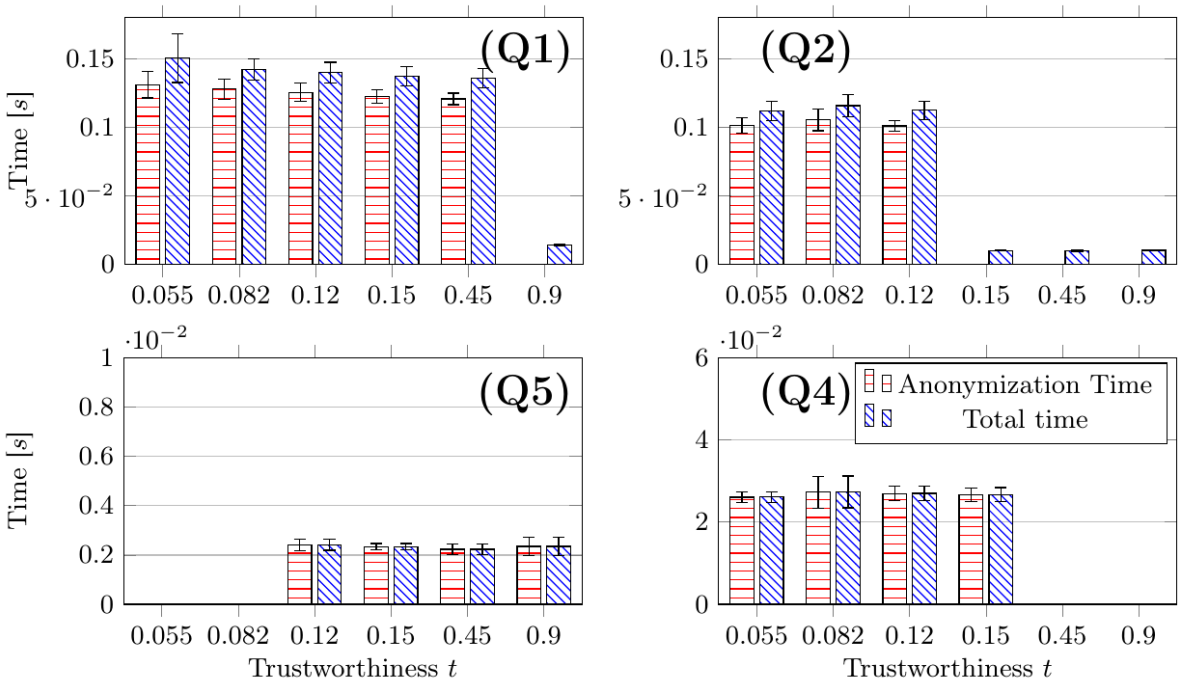


Figure 5.7: Average anonymisation time (horizontal striped bars) and average total response time (diagonally striped bars) for **Q1**, **Q2**, **Q4**, and **Q5** (data-views) and 6 different users (trust levels).

and $k = 6$),the anonymization (when needed) still impacts the performance in the same scale then **Q1** and **Q5** with very low anonymity level (respectively $k = 2$ and $k = 1$).

From these experiments, we observe that when anonymization is applied the response time increases, but, even in the worst cases, the increase is far less than one order of magnitude, and, basically, it has no impact on the real-time response of the system. Moreover, the application of different levels of anonymization (different k in our case) has a small impact. We will investigate in the next paragraph the effect of the data-view size on the Anonymization and Response time.

Let us analyze the behavior of the anonymization time increasing the size of the dataset. Typically patterns run in the limited time window (e.g., 10 to 30 minutes) producing small-sized data-views (i.e., in the range of $10 - 10^3$). To investigate the scalability of our approach, in Figure 5.8, we report the average anonymization time variation for 5 different data-view {**Q1** to **Q5**} (with 5 different sizes see Table 5.10) and a low trustworthiness level ($t = 0.055$, so anonymization is always applied). As mentioned above, the worst case (around $5 \cdot 10^4$ records) takes less than $150ms$, and a linear extrapolation of the data allows as to estimate the anonymization time for a 10^5 data view (so, 100 times the typical size) around $200ms$, which it can be safely considered as a real-time response for our use case.

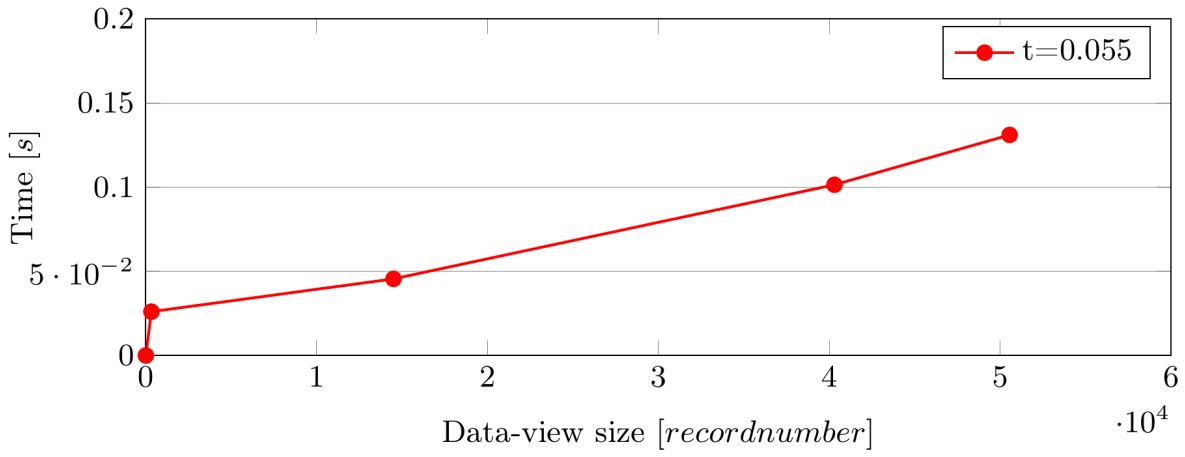


Figure 5.8: Average anonymization time variation according to data-view sizes (for trustworthiness $t = 0.055$).

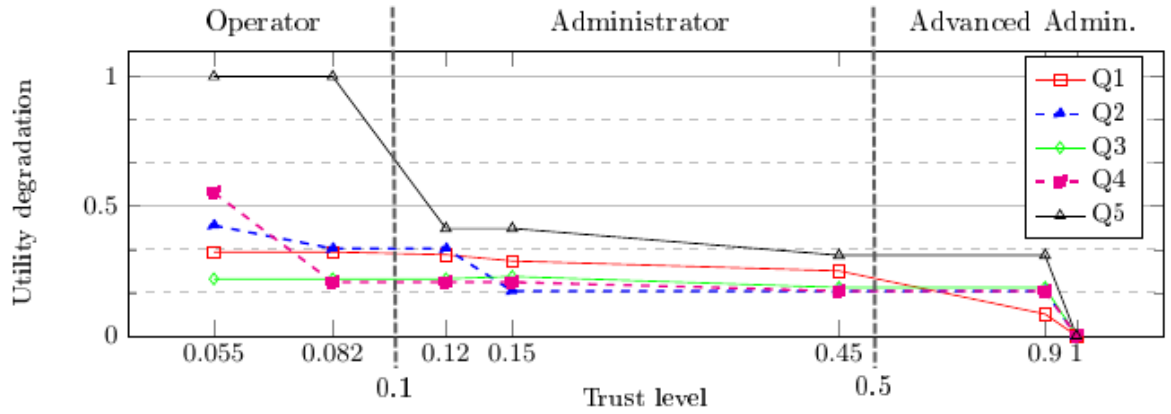


Figure 5.9: Utility degradation by trust level for different queries

Utility: Trustworthiness levels (i.e., risk threshold) should be set to allow the best a trade-off between data exploitation and privacy protection. In our use case we set our trustworthiness levels respecting a conventional distribution of privacy risk levels presented in Table 5.11, and we would like to investigate the convenience of this repartition by answering the following question: Do these trustworthiness levels provide enough data (or data with enough utility) to allow each user/role to fulfill their tasks described in Table 5.8. In Figure 5.9, we report the utility degradation according to the six selected trustworthiness levels, representing the 3 roles (reported on the top of the figure). We

can observe that the utility degradation (obviously) decreases as we increase the trust level, with the limiting case of $t = 1$ with no utility loss (and no anonymization) for the Advanced Administrator. For most of the patterns (4 over 5, so except **Q5**), the Operator role has a maximum utility loss of 30%, showing that the specific anonymization transformations applied are strongly decreasing the risk, and limiting the impact on the utility. That should allow performing the analysis on the anonymized data, without the need to enhance the trust level (so no need to get Admin rights).

In the case of **Q5**, the anonymization is not able to significantly decreases the risk, without largely impacting the utility. In fact, the Operator is left with no information (utility degradation = 1), and to analyze the result an increase of the acceptable risk threshold (trust level) is needed. Enhancing trust (i.e. assigning Admin rights to the Operator) could reduce the utility degradation in the $30\% - 40\%$ range, likely allowing the assessment of the pattern result. We should note, that **Q5** is particularly hard to anonymize, because it has fewer events (around 10), and, since k-anonymity is a measure of indistinguishability, it needs strong anonymization.

Figure 5.9 also shows that in most cases increasing the trust level for Administrator or even Advanced Administrator (except of course for $t = 1$, where we have no anonymization) the impact on utility degradation is moderate: for example **Q1** and **Q4** are almost flat in the Administrator zone, similarly **Q2** has a first drop, and stays flat in the Administrator and Advanced Administrator parts. In other words, increasing the risk thresholds, we could take more risk, but we do not gain much in terms of the utility. This counter-intuitive effect is mostly due to the difficulty to find an anonymization strategy able to equalize the risk threshold. As mentioned in Section 5.2.3, in practical cases the number of possible anonymization strategies is limited, and to fulfill the condition of Eq. 4.1 the final risk may be quite below the risk thresholds (trust values). In practice, in many cases, even increasing the risk thresholds (trust values), it is not possible to find a more optimal (from the utility point of view) anonymization strategy. In Figure 5.9 we show the utility loss for four patterns both showing the risk thresholds (dotted lines) and the *actual* risk achieved after the anonymization. In the ideal case, the two curves should be the same, meaning that we could always find a transformation that equalizes actual risk and risk thresholds (trust), but in practice, we see that we are often far from this optimal condition. For example, for pattern **Q2**, with risk thresholds $t = 0.15$, $t = 0.45$ (Administrator role) and $t = 0.9$ (Advanced Administrator), indicated with red circles, we have the same value of utility degradation. In fact, the anonymization strategy found for $t = 0.15$ case, corresponds to an actual risk of 0.14 (square dots with a circle in Figure 5.9, upper-right panel), so quite close to the threshold. Increasing the thresholds to $t = 0.45$ and $t = 0.9$ (round dots with a circle in the figure), no better strategies were found, so the same anonymization strategy is applied, and clearly, the final risk is still 0.14 (and utility is the same), well below the thresholds. Similar effects are also present in the other

patterns.

The experimental analysis shows that adapting the anonymization to the specific patterns, we can mostly preserve enough information for the investigation, keeping the privacy risk low. In cases where this is not sufficient, typically characterized by small data set, the trust enhancement strategy can support the access to less-anonymized data.

5.7 Policy Implementation

In this section, we present a possible way of expressing the authorizations model, described in Section 4.3 in Eq.4.1, through risk-based policies.

Since we based the access control model and architecture on a modified version of XACML's architecture, we will also propose an extended version of XACML's language to implement our policies (attribute based policies).

The Policy sample we present in this section will implement the authorizations presented in the use case "*privacy aware threat investigation*" introduced in Section 5.3. We propose to organize policies according to *patterns*, i.e., each policy expresses the authorizations required to run and investigate a pattern. This choice allows for a flexible policy management (addition, modification, and deletion), in fact, new patterns are often added and old once updated or removed from the system and each of these operations requires an update of the policies. Hence if we dedicate a policy for each pattern, when an existing pattern is modified or a new one is created, we just need to revise the policy expressing the authorizations required by the pattern or create a new policy.

The proposed policies have a similar structure as described in Example 5.1. In this policy example, we express the authorizations required by the pattern detecting *Brute Force Attacks*.

```
<!-- Brut_force_attack policy -->
<Policy PolicyId="brut_force_attack_policy" RuleCombiningAlgId="permit-overrides">
    <Target> ... </Target>
    <Rule RuleId="deny_all" Effect="Deny">
    <!-- Deny access for the policy Target -->
    ...
    <Rule RuleId="allow_trust_higher_than_risk" Effect="Permit">
    <!-- Allow access if trust>=risk -->
    ... </Rule>
    <Rule RuleId="adjust_risk_higher_than_trust" Effect="Permit">
    <!-- Adjust Trust or Risk Values if trust<risk -->
    ... </Rule>
    <Rule RuleId="exceptional_rule_1" Effect="Permit">
    <!-- optional -->
    <Target> <!-- exception's target --> </Target>
</Rule>
</Policy>
```

Listing 5.1: Risk-Based Policy Sample: Brute Force Attack Pattern

Each policy is composed by a main **target** and three **rules** expressing the three possible outcomes of the access evaluation, i.e., *deny*, *grant*, or apply an adjustment strategy

σ (see Eq 4.1 in Section 4.3). Rules will apply to the same target defined as the policy target. We use the combination algorithm *permit-overrides* to select the rule to enforce, in case more then one rule is applicable (i.e., rule's target matches the request's target and the rule's conditions are satisfied by the request). The algorithm *permit-overrides* enforces the first rule that permits access (allows access) once the obligations defined by the rule are enforced (see [58] for more details). If no rules permit access that the first rule denying access will be enforced. Some patterns may require additional *exceptional* rules (i.e., break the glass rules) usually more permissive and with a more specific target. For instance, for very critical patterns (e.g., a denial of service attack) we can allow a super administrator to access the data for investigation without trust assessment despite the risk level (see Example 5.6).

The main target describes which subjects (requesters), which resources and what actions the policy applies to. In example (Example 5.1), for instance, the policy applies to any requester with any role known to the risk-based access control system *ROLE:ANY* (see Example 5.2). The targeted resource is the table containing the logs *TABLE:LOGS* and the targeted action is *ACTION:READ* access. The policy target should also specify the context of the access which can express for instance the access purpose e.g., *running* or *investigating* the pattern *PATTERN:BF-ATTACK*. It could be also used to describe the security context e.g., *alert* or *logged event*. The policy target (or main target) can be refined within the rules. For instance, the exceptional rule target (Example 5.6) refers to the group of subject with the role of *ROLE:SUPER_ADMIN* which is more specific than *ROLE:ANY*, it also narrows the context and make the rule only applicable in case of a security alert *SEC_CONTEXT:ALERT*.

```
<Target>
    <AnyOf> <AllOf> <Match MatchId="string-equal">
        <AttributeValue DataType="string">ROLE:ANY</AttributeValue>
        <AttributeDesignator AttributeId="subject-role" Category="subject" .../>
    </Match> </AllOf> </AnyOf>
    <AnyOf> <AllOf> <Match MatchId="string-equal">
        <AttributeValue DataType="string">TABLE:LOGS</AttributeValue>
        <AttributeDesignator AttributeId="resource-id" Category="resource" .../>
    </Match> </AllOf> </AnyOf>
    <AnyOf> <AllOf> <Match MatchId="string-equal">
        <AttributeValue DataType="string">ACTION:READ</AttributeValue>
        <AttributeDesignator AttributeId="action-id" Category="action" .../>
    </Match> </AllOf> </AnyOf>
    <AnyOf> <AllOf> <Match MatchId="string-equal">
        <AttributeValue DataType="string">PATTERN:BF-ATTACK</AttributeValue>
        <AttributeDesignator AttributeId="pattern-id" Category="environment" .../>
    </Match> </AllOf> </AnyOf>
</Target>
```

Listing 5.2: Policy main target

The first rule, *Deny Rule* in the Policy (Example 5.3) does not specify the target, hence, inherits the policy's target. This rule does not have any conditions or obligation its aim is to guarantee that the access is denied if none of the other rules allow it (e.g., the

conditions of other rules were not satisfied or errors occurred during the rules evaluation).

```
<Rule RuleId="deny_all" Effect="Deny">
    <!-- Deny any access to all Roles/Subjects in the Target -->
</Rule>
```

Listing 5.3: Deny Rule

The *Adjust Rule* (in Example 5.4) applies to the policies target as well. It expresses the second outcome in Eq. 4.1, where an adjustment strategy σ is required to be applied before granting access. The condition of application of this rule (in Line 3), is that the trust is lower than the risk level. To check this condition, we need, first, to compute the trust and risk values, which as indicated in Section 4.4, is the task of the *RBA-IP*, however we need to indicate to the *RBA-IP* where to possibly find the information, e.g., we define that the *request-trust* can be computed using the *TrustAssessmentModule* (Line 8) and the *request-risk* can be computed by the *RiskAssessmentModule* (Line 15). In case of failure to compute the trust level, minimum trust level $T = 0$ will be assigned to the request, and if the failure occurs in the risk computation, we will assign to the request maximum risk level $R = 1$. If the condition of *Adjust Rule* is fulfilled then access cannot be granted to the resource, unless an adjustment phase is successfully carried. The adjustment strategies for each pattern are expressed through obligations (see Example 5.7, 5.8, or 5.9.).

```
1  <Rule RuleId="adjust_trust_lower_than_risk" Effect="Permit">
2  <!-- Ajust Trust or Risk Values if trust < risk -->
3      <Condition><!--  applicable if trust is lower than risk -->
4          <Apply FunctionId="double-greater-than-or-equal">
5              <Apply FunctionId="function:or"> <!-- compute the trust or trust =0 -->
6                  <!-- call TrustAssessmentModule to compute the trust -->
7                  <Apply FunctionId="function:double-one-and-only">
8                      <AttributeDesignator Category="request-trust" AttributeId="trust" Issuer="
       TrustAssessmentModule"/>
9                  </Apply>
10                 <AttibuteValue>0</AttributeValue>
11             </Apply>
12             <Apply FunctionId="function:or"> <!-- compute risk or risk =1 -->
13                 <!-- call RiskAssessmentModule to compute risk -->
14                 <Apply FunctionId="function:double-one-and-only">
15                     <AttributeDesignator Category="request-risk" AttributeId="risk" Issuer="
       RiskAssessmentModule"/>
16                 </Apply>
17                 <AttibuteValue>1</AttributeValue>
18             </Apply>
19         </Apply>
20     </Condition>
21     <ObligationExpressions>
22         <!--Adjustment Strategies-->
23     </ObligationExpressions>
24 </Rule>
```

Listing 5.4: Adjust Rule

The *Allow Rule* (see Example 5.5) expresses the last out come of evaluation in the authorization model (Eq.4.1). Similarly to the *Deny Rule* and *Adjust Rule*, this third rule, has the same target as the policy. According to *Allow Rule*, access to the requested data is fully granted if the trustworthiness level of the request is higher than its risk level.

Table 5.12: Obligation Types

at-decision Obligations	Are similar to the classic XACML3.0 obligations they are actions to be enforced at the same time then the access decision e.g., sending notifications, logging session details. These obligations fulfillment do not influence the access decision
pre-decision Obligations	Are actions to be enforced before enforcing the access decision to a resource e.g., anonymization, encryption, requesting a stronger authentication. The success or failure to fulfill these obligations can influence the access decision
post-decision Obligations	Are actions expected to be enforced after enforcing the access decision e.g., deletion of the data

The trust and risk levels assessment is expressed the same way as the *Adjust Rule* in Example 5.4 (Lines 4 to 18).

```
1 <Rule RuleId="allow_trust_higher_than_or_equals_risk" Effect="Permit">
2 <!-- Allow access if trust >= risk -->
3     <Condition><!--  applicable if trust is higher than or equals risk -->
4         <Apply FunctionId="double-greater-than-or-equal">
5             <!-- compute and compare trust and risk levels -->
6         </Apply>
7     </Condition>
8 </Rule>
```

Listing 5.5: Allow Rule

Obligations, in the XACML standard, are enforced by the PEP immediately *after* granting or denying access, e.g., allowing access to a user *Alice* with the obligation to log *Alice*'s actions during the access session. However, our authorization model needs, in some cases to enforce certain actions *before* granting access, such as transformations on data, and other actions during the consumption of the data. Thus we propose to use two other types *pre-decision* and *post-decision* obligations categories. These new obligation categories were inspired by [163]. In Table 5.12 we provide a description for each category, we also provide an implementation example in Examples 5.7, 5.8 and 5.9.

We discussed in Section 5.6.5 possible ways to dynamically select the mitigation strategies based on utility, but we are not including the implementation in this paper.

```
1 <Rule RuleId="exceptional_rule_1" Effect="Permit">
2 <Target>
3     <AnyOf> <AllOf> <Match MatchId="string-equal">
4         <AttributeValue DataType="string">ROLE:SEC_ADMIN</AttributeValue>
```

```
        <AttributeDesignator AttributeId="subject-role" Category="subject" .../>
    <AnyOf> <AllOf> <Match MatchId="string-equal">
        <AttributeValue DataType="string">SEC_CONTEXT:ALERT</AttributeValue>
        <AttributeDesignator AttributeId="pattern-id" Category="environment" .../>
    </Match> </AllOf> </AnyOf>
</Target>
</Rule>
```

Listing 5.6: Exceptional rule allowing the super admin to access without risk and trust assessment

```
<ObligationExpression ObligationId="system:log" ObligationType="at-access" FulfillOn=Permit>
    <!-- Temporarily Grant higher trust level -->
    <!-- Log the access request and access session -->
</ObligationExpression>
```

Listing 5.7: at-access Obligations

```
<ObligationExpression ObligationId="system:anonymize" ObligationType="pre-access" FulfillOn=
    Permit>
    <AttributeAssignmentExpression><!--compute required anonymity level-->
        <AttributeDesignator AttributeId="optimal_k" Issuer="TrustAndRiskAjustementModule" />
    </AttributeAssignmentExpression>
    <AttributeAssignmentExpression><!-- apply anonymization -->
        <AttributeDesignator AttributeId="anonymizer:k-anonymity" Issuer="
    TrustAndRiskAjustementModule" />
    </AttributeAssignmentExpression>
</ObligationExpression>
```

Listing 5.8: pre-access Obligations

```
<ObligationExpression ObligationId="remote-rba-ep:data-deletion" ObligationType="post-access"
    FulfillOn=Permit>
    <AttributeAssignmentExpression><!-- Enhance Trust-->
        <AttributeDesignator AttributeId="enhanced-trust-level" Issuer="
    TrustAndRiskAjustementModule"/>
    </AttributeAssignmentExpression>
    <AttributeAssignmentExpression><!-- fix access time window-->
        <AttributeDesignator AttributeId="time-window" Issuer="TrustAndRiskAjustementModule"/>
    </AttributeAssignmentExpression>
    <AttributeAssignmentExpression><!-- delete data-->
        <AttributeDesignator AttributeId="action:data-deletion"/>
    </AttributeAssignmentExpression>
</ObligationExpression>
```

Listing 5.9: post-access Obligations

5.8 Chapter conclusions

Nowadays preserving privacy is a major concern in every organization. Software market leader seeks to develop a new efficient solution to address privacy and security issues and provide innovative products offering the best tread-off between privacy preserving and data exploitation.

In this chapter, we propose a privacy-aware risk-based access control model able to address these issues and support a flexible access control in privacy demanding scenarios. In our model, we propose to manage the privacy-risk using the concept *syntactic anonymity*. This category of privacy metrics is designed to ensure privacy-protection in

data publishing [43]. Although these metrics have received a fair amount of cretinism, they are still widely used. For instance, when publicly releasing sensitive datasets, using such metrics will clearly lower privacy risk compared to releasing actual data values. In addition syntactic anonymity is preferred in some cases since it provides better quality of data than most noise addition techniques like differential privacy.

We focus (in this chapter) on *re-identification* risk assessed using a well known metric k-anonymity. However, the approach is not bound to these choices and it can be readily adapted to alternative syntactic metrics (e.g., ℓ-diversity, t-closeness). It can also integrate the concept of differential privacy to which we will dedicate the next chapter (Chapter 6).

When the privacy risk is too large, the framework can apply privacy preserving adjustment strategies (risk mitigation and trust enhancement strategies) to increase the exploitation of the data while ensuring an acceptable risk level. As an example of privacy risk mitigation, we propose to apply on-the-fly anonymization, instead of denying access to any "risky" information, The level of anonymization is dynamically assessed (for each data request) to enhance the availability of information while respecting a privacy level desired by the organization. Current anonymization techniques are typically computationally intensive [84, 164] and their applicability is limited to off-line scenarios or small size datasets, diminishing their business impacts, not allowing the usage by more advanced applications, such as real-time analytics and on-demand data services. In practice, with current technologies, querying a large database and extracting an anonymized dataset in real-time is not possible, and most anonymization processes are run off-line (i.e., as batch processes). However new recent technologies (such as in-memory databases combined with column-store optimized algorithms) would facilitate the implementation of run-time data anonymization, allowing our model to be easily integrated with new data-intensive business applications.

In the experimental sections of this chapter, we show how the framework can simultaneously address both the privacy and the utility requirements within different industrial use cases. Indeed the obtained results show that the framework leads to meaningful results and real-time performance.

Chapter 6

Differential privacy based access control

Syntactic anonymity heuristics propose to alter identifying attributes to grantee certain level of anonymity. Although this approach presents some weaknesses, it is still widely applied especially when publicly releasing sensitive data-sets. Another privacy-preserving approach, the differential privacy, have been lately proposed as an alternative to syntactic anonymity. Differential privacy is a formal mathematical framework allowing to guarantee a certain level of privacy when analyzing or releasing statistical data.

In this chapter we propose a different version of the privacy-aware risk-based access control model, based on differential privacy and more suitable to preserve privacy in the context of data mining. The model allows for data access at different privacy levels, generating an anonymized data set according to the privacy clearance of each request. The architecture also supports re-negotiation of the privacy level, in return for fulfilling a set of risk and trust adjustment strategies expressed through Access and Usage Control Obligations. We also show, how the model can address the privacy and utility requirements, in an human-resource motivated use-case with a classification task. The model provides a flexible access control, improving data availability while guaranteeing a certain level of privacy.

6.1 Introduction

In Chapter 4 we proposed a novel access control model aiming, like most risk-based, to bring more flexibility, replacing (or integrating) pre-defined access control policies, with access decisions based on the risk estimation of specific requests. Our model evaluates each request using a user/role dependent risk and risk thresholds which can be set considering

the trustworthiness of a requester or a request's context. Our approach goes beyond the binary access decision (allow, deny) supported by most risk-based models, and we propose a third option, which is to adjust the trust and risk levels allowing for restricted, partial and/or monitored access to the data instead of denying risky requests.

This approach requires domain-specific risk assessment heuristics. In Chapter 5 we propose to explore how to apply the approach in the context of data privacy. Therefore we develop a privacy-aware adaptation of our framework using syntactic anonymity metrics (e.g., k-anonymity) to assess the risk (i.e., privacy-risk) and privacy-related factors to assess trust. Although widely used in practice, k-anonymity and the related family of syntactic privacy metrics [43]), is susceptible to various attacks (e.g., [62]), and, in the last 10 years, another formal approach has been proposed to provide strong privacy guarantee: differential privacy [54].

In this chapter, we propose a privacy-aware risk-based access control model, which uses differential privacy to reduce the data disclosure risk. The model, in case the access to raw data is not permitted, is able to provide a differential private data set, according to the *privacy clearance* of the user, which plays the role of the trust in the previous model. This allows for a more flexible access, improving data availability, and at the same time, guaranteeing a formal level of privacy.

The main contributions of this chapter are as follows:

1. We propose a privacy-aware risk-based access control model that evaluates access and clearance decisions based on a privacy-preserving approach.

2. We propose to use a differentially private algorithm to enforce these decisions, respecting the adequate privacy level.

3. We define an architecture for our access control system, which integrates a classic policy-based access control, and also supports mechanisms for (temporarily) increasing privacy clearance.

4. We implement a proof-of-concept prototype and run preliminary experiments, to evaluate the utility of the data, using a simple classification task, and the performance of the system.

In the next section (Section 6.2), we provide a motivating use case for our work. In Section 6.3, we give a short overview on Differential Privacy. In Section 6.4, we introduce our privacy-aware access control model. Section 6.5 is dedicated to the description of the architecture of the access control framework. In Section 6.6, we describe the experimental evaluation and discuss the main results. We conclude, in Section 6.7,with some final remarks.

Table 6.1: Usage scenarios, comprising different actors (data requesters), security levels, and expected utility (i.e., type of reports needed)

#	Role	Operation	Risk	Utility
1	HR manager	HR view (int.)	Low Risk	full access
2	HR manager	HR view (ext.)	Medium Risk	aggregated
3	HR developer	Testing data	Medium-High Risk	anonymized
4	HR Benchmarking	Benchmark	High Risk	anonymized

6.2 Use Case

Human Resource (HR) data are becoming increasingly important for the management of the company workforce. Whereas traditionally, they were mostly accessed in tabular form from the HR department and people managers, there is nowadays a large number of additional analytics and functionalities to improve HR key processes [100, 142] (e.g., talent discovery, compensation process, training), and, correspondingly, there is an increased need for access to HR data, reports and analytics, involving multiple actors in the company. At the same time, HR data contain sensitive and personal information, which is subject to, often complex, data protection regulations, and data access should carefully manage.

For example, an HR manager can have a full view of the HR information for her/his department, but an aggregated view for the HR data from other departments. In some cases, for example, employee survey results for collecting employee feedback, a certain level of anonymity is needed even for the data within the department. Legal framework, such as the European data protection regulations [59], can additionally impose geographical constraints on access and transfer of personal information.

HR data are also needed for the testing phase in the development of HR applications. In this case, the data should not contain personal information, but they should be realistic *enough* to allow for significant testing. So, an in-house developer may have access in a controlled environment to an anonymized version of the data. If the development task is outsourced to an external company, an even stronger anonymized is likely needed.

HR data (e.g., compensation and health care cost data) are also sometimes shared with external parties for benchmarking purpose (see e.g. Bureau of Labor Statistics (BLS) [132]). and, for that scope, they need a high level of privacy guarantees to be released.

The requirements of these illustrative examples can be summarized as in Table 4.1. These scenarios show how a rather complex access control framework should be set up to address the privacy requirements. Currently, in most cases, these requirements are addressed with a mix of specific configurations of traditional access control systems (e.g, RBAC systems for the HR manager use-case), usage of specific anonymization tools (e.g.,

for releasing data for application testing or benchmarking services), and, often, relying on human-based processes. In the next sections, we will show how these scenarios can be realized.

6.3 Background on Differential Privacy

Differential privacy [54] is a privacy framework devised for providing a formal, strong privacy guarantee. Whereas, traditionally, privacy-preserving data publishing was based on syntactic privacy [43] mechanisms, where, for example, it is imposed as condition that a record being indistinguishable from k other records [135] (equivalence group), or the sensitive values to be *well* distributed within the equivalence groups [92, 99], differential privacy takes another approach, requiring that the answer to any query being *probabilistically indistinguishable* if a particular record is present in the database or not. In other words, an adversary cannot learn (almost) anything about an individual record, since the output does not (almost) change, whether that specific record is present or absent in the data set. Following [113], we can define differential privacy, in the context of privacy-preserving data publishing, as:

definition 6.3.1. A randomized algorithm $\mathcal{K}$ satisfies ϵ-differential privacy if for all pairs of data sets D, D', differing for at most one record ($D \sim D'$), and for all possible anonymized data sets $\hat{D}$, we have that:

$$Pr[\mathcal{K}(D) = \hat{D}] \leq e^{\epsilon} \times Pr[\mathcal{K}(D) = \hat{D}]$$

where the probability is computed over the randomness of $\mathcal{K}$, and the parameter $\epsilon > 0$ sets the bound of the privacy guarantee, with low values of ϵ providing stronger privacy.

The mechanism for providing differential privacy (called ϵ-differentially private sanitizer) is typically based on noise addition. There are two approaches: interactive and non-interactive. Historically, differential privacy was devised for the interactive model [54]: a user sends a set of queries to a database, and the database owner, to assure privacy, adds some random perturbation to the query answer (e.g., adding Laplace noise with variance related to ϵ parameter). Although the interactive framework is mostly used, it has some drawbacks [146], e.g., after a limited number of queries the noise level should be increased, highly impacting the utility.

In the non-interactive model (see [91, 94, 113, 146]), the database owner anonymizes the original raw data, and then releases the anonymized version, providing the user a greater flexibility for data analysis, and basically no limitation in terms of queries. Indeed this model allows producing data-views meeting a certain level of privacy and utility. However, it is important to mention that this assumption does not take multiple releases in consideration, and the utility is estimated considering a generic assumption on what

the data will be used for (e.g., assuming that the data will be used analysis relies mostly on the counts of certain attributes).

We will be using the latter model in this chapter. In particular for deriving differential private data set for our evaluation (see Section. 6.6), we follow the approach of [113]. The method considers the raw data, and it computes the contingency tables, counting the number of records sharing a combination of attributes. Then, it probabilistically (using an exponential mechanism) generates a generalized contingency table (generalizing attribute values in wider classes). Then, it applies Laplacian noise to the generalized contingency table. The generalization step allows increasing the counts for the cells, resulting in lowering the utility-impact of the noise addition. Synthetic data can be produced from the generalized and randomized contingency table. The resulting data set, generated by a ϵ-differential privacy mechanisms, can be safely used for any data analysis (we will test it on a classification task, as in [113]).

6.4 Differential privacy based access control model

In this section, we provide a general description of our differential privacy based model. The privacy-aware risk-based access control model presented in this chapter can generally be mapped to the general model presented in Chapter 4. However, unlike the model based on syntactic anonymity (presented in Chapter 5), it is harder to quantitatively assess the privacy-risk (before mitigation) using differential privacy since this privacy metric quantifies the privacy guarantee provided by a release mechanism, and it uses added noise to achieve this measure.

Therefore, to assess the privacy-risk in this model, we propose a qualitative risk measure based on the sensitivity of the data requested and the presence of identifying attributes. The application of differential privacy requires a limited number of queries and a knowledge about these queries, this makes easier the task of assessing the privacy-risk. We consider here a risk domain of 5 qualitative risk levels $\mathcal{L} = \{l_1$: Very-Low, l_2: Low, l_3: Medium, l_4: High, l_5: Very-High$\}$. Note that the number of levels can vary depending on the use-cases and risk policies, with no major differences for the access control model. In Table 6.2 we map a required privacy clearance interval $(\mathcal{T}_1 \dots \mathcal{T}_5)$ with each of risk levels based on a tread-off between the expected utility (see Table 6.1) and the privacy goals of the company. In the last column (of Table 6.2) we report the required level of sanitization ϵ_i corresponding to the clearance $\mathcal{T}_i$.

The authorizations in our model can be built as an extension of the traditional policy-based authorizations, such as XACML model [115]. Generally speaking, the model proceeds as follows: The policy determines the clearance level/interval (depending on its risk level) required to access a given dataset (data resulting from a given query). Whenever a user/role issues a request, the access control model checks if his/her request has

Table 6.2: Mapping between privacy risk, required privacy clearance and equivalent level of sanitization

	Privacy-risk	**Privacy Requirement**	**Needed sanitization**
l_1 :	Very-Low risk	$\mathcal{T}_1 \approx 0$	*none* ($\epsilon_1 \approx +\infty$)
l_2 :	Low risk	$\mathcal{T}_2 = [0.01, 1[$	ϵ_2 $]1, 10]$
l_3 :	Medium risk	$\mathcal{T}_3 = [1, 10[$	$\epsilon_3 \in]0.1, 1]$
l_4 :	High risk	$\mathcal{T}_4 = [10, 20[$	$\epsilon_4 \in]0.05, 0.1]$
l_5 :	Vary-High risk	$\mathcal{T}_5 = [20, +\infty[$	$\epsilon_5 \leq 0.05$

the adequate privacy clearance (with respect to the clearance level/interval of the issued query).

Differently from the classic policy-based access control, the system, in addition of a allow or deny decision, can deny access to the data set in the raw version, but still, provide the user with an anonymized version of the data.

More formally, each access query $req = (u, v)$[1] is characterized by a privacy clearance T_ϵ which depends on user u and context information C (e.g., within the corporate network users may have a larger clearance); a required clearance level $\mathcal{T}$ assessed depending on the risk level $risk$ of the dataset v resulting from the query.

The query req with a $risk = l_i$ will be evaluated by the function $Auth(req)$ defined as follows:

$$Auth(req) = \begin{cases} \mathsf{grant} & \text{if } T_\epsilon \in \mathcal{T}_i \\ \mathsf{adjust}_{\sigma i} & \text{if } T_\epsilon \leq \mathcal{T}_i \\ \mathsf{deny} & \text{if } denied_\Pi \end{cases} \tag{6.1}$$

T_ϵ the privacy clearance of the request, expresses the trust we have that a query will not violate privacy it plays the role of trust parameter used in the general model (Chapter 4 Section 4.3). $T_\epsilon \equiv T_\epsilon(u, C)$ depends on user u and context information C (e.g., within the corporate network users may have a larger clearance)

Note that the privacy clearance parameter T_ϵ, here, plays a role similar to the privacy budget [126] typically used for differential privacy models. But, in our case, we only consider accessing disjoint sets of data, so each user/role can *spend* all his/her budget for a single request, and he/she has access to data at the same, or lower, level as the privacy clearance. This is similar to the security clearance parameter in multi-level security models.

Therefore, a user u (say the people manager of the department) with a trust clearance $T_\epsilon(people\_manager)$ is granted access to an dataset v (say the HR data of a de-

[1] In most cases the dependency between u and v is mediated by roles and/or permissions that should also be considered in the evaluation of the query. However for the sake for simplicity, we do not consider roles, and focused only on read access, for an extension of this model including roles, we can follow the lines of access control risk models as described in [14, 34].

partment $risk = l_i$) if the access control policy of the dataset Π includes user u and $T_\epsilon(ppl\_manager) \in \mathcal{T}_i$ the clearance required by v.

If the query clearance is not sufficient $T_\epsilon \leq \mathcal{T}_i$ we propose to apply the adjustment strategy σ_i. An adjustment strategy σ_i can be predefined for each risk level l_i[2], to mitigate the risk (e.g. by sanitizing the data) and/or increase the privacy the same way we propose to enhance trust in the general model (in Chapter 4 Section 4.3).

If we chose to provide a sanitized version of the data (say for the people managers of other departments), in this case, the system retrieves the *privacy clearance* value, T_ϵ, associated to the user/request, and it applies the differentially private sanitizer to the original data to obtain a data set of differential privacy $\epsilon = 1/T_\epsilon$[3].

Adding this option of providing sanitized data can increase the flexibility and, ultimately, the access to data. On the other hand, especially for large privacy clearances, T_ϵ, the high level of sanitization (very small ϵ) can severely impact the utility, making the data not usable. To this aim, we foresee mechanisms to (temporarily) increase the privacy clearance to meet the expected utility, for example asking the user to fulfill some obligations (as we proposed in Chapter 4). The architecture described in Section. 6.5 can support this privacy clearance enhancement (trust enhancement) functionality, but, we do not discuss them in details in this study, focusing more on data sanitization for risk mitigation.

Finally Access can be denied if the policy denies[4] $\Pi(req) = denied$. Of course, this implies that for the first two outcomes (allow and adjust) the request was not denied by the policy.

6.5 Architecture

In this section we present an abstract architecture for our privacy-aware Risk-based access control framework based on differential privacy. We will also highlight the main modifications with respect to the architecture of the general model presented in Chapter 4 Section 4.4 The architecture, depicted in Figure 6.1, is composed of three main modules:

Privacy-Aware Access Control Module is the entry point of the system, through which users can submit requests to retrieve data from the underlying database. This module evaluates the access request, and it grants access to (original or sanitized version of) the requested data or denies access.

For this scope, the Privacy-Aware Access Control Module assesses the data request

[2]The possibility of dynamically selecting an adjustment strategy was also discussed in Chapter 5

[3]Note that the system may have already in the cache the anonymized data set, if it had received the same data request at the same privacy clearance. In this case, there is no need to re-anonymize the data, and it uses the already produced data set, improving performance and security.

[4]explicitly or implicitly as discussed in Chapter 4

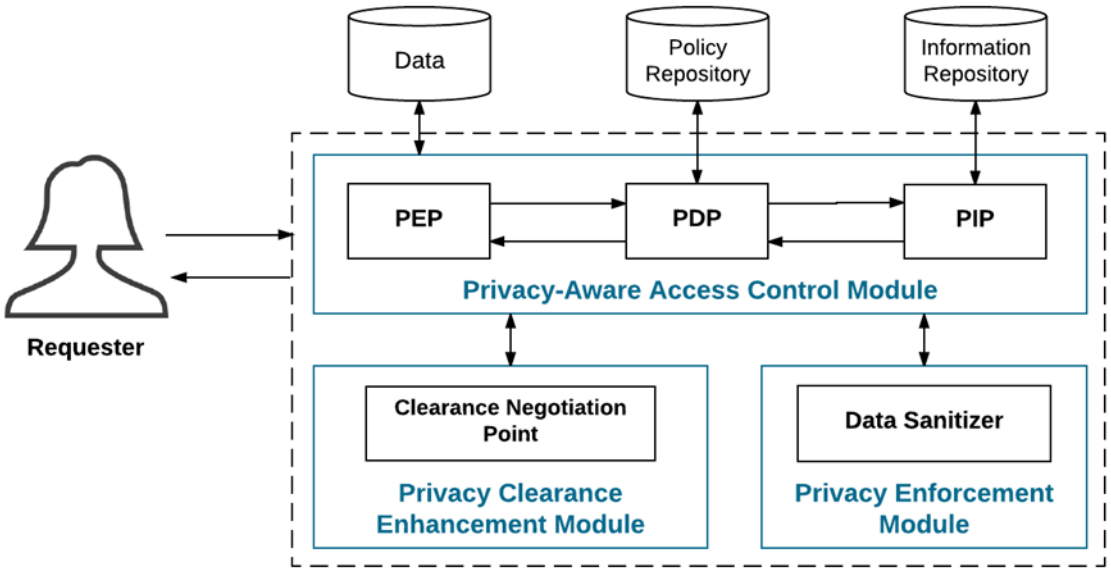


Figure 6.1: Architecture of the Privacy-Aware Risk-Based Access Control framework (Based on differential privacy)

against an access policy to determine whether the requester has the needed authorizations to access the resource (requested data-view) and, also, to evaluate the privacy clearance (as discussed in Section. 6.4). Then, the decision is enforced by calling the Privacy Enforcement Module or renegotiate by calling the Privacy Clearance Enhancement Module.

The Privacy-Aware Access Control Module is based the on the XACML (eXtensible Access Control Markup Language) standard [115]. XACML is a declarative fine-grained, access control policy language. The standard also provides an access control architecture and a description of the access evaluation process (data-flows, access request, access decision etc.)

In this module Access Control is realized internally using a PEP-PDP [5] pair. A PIP (Policy Information Point) is used to provide additional information needed to evaluate the request and estimate its privacy clearance (e.g., in our use case if the requester is a manager, we would like to know her/his department in order to define her/his privacy clearance, if the requested data contains information about his department this queries clearance will be higher than the clearances of queries requesting data about other departments)

[5] In XACML the PDP is the point that evaluates an access request against an authorizations policy and issues an access decision and the PEP Policy Enforcement Point is the point that intercepts user's request, it calls the PDP for an access decision then it enforces the decision by allowing or denying the access.

Privacy Enforcement Module. After evaluation of the access request, the Privacy Enforcement Module receives a data view (non-anonymized version) and a privacy clearance value. The role of this module is applying data sanitization algorithms, and generating an anonymized version of this data view, according to the privacy clearance.

Privacy Clearance Enhancement Module. The privacy clearance defined by the Privacy-Aware Access Control Module can be re-negotiated to a higher level in some cases (for example if the utility of the anonymized data is not sufficient) to allow more flexibility. The user can ask (temporally) for a higher clearance, in exchange, for example, of fulfilling some obligations to mitigate the additional risk. These operations are typically expressed as access and usage control obligations (see Chapter 4), for example imposing deletion of a resource after that a retention period expires, or providing stronger authentication credentials.

It is easy to see that the Privacy-Aware Access Control Module is also risk-based in this architecture and can be mapped to the *Risk-based access control module* in the general architecture presented in Chapter 4, Figure 4.2. The Privacy Clearance Negotiation Moduleplays the role of *Trust Enhancer* aiming to provide the requester/request with higher privacy clearance to reach the required clearance to execute a query. The Data sanitizer in the Privacy enforcement Module allows enforcing the selected risk mitigation strategy. Combined together the Privacy Clearance Negotiation Module and Privacy enforcement Module can be mapped to the Trust and Risk adjustment module in the general model Figure 4.2. No *Risk estimation* or *Trust estimation* modules are included in our architecture (Figure 6.1) since the version of the model presented in this chapter do not support dynamic risk and trust (trust is substituted by the privacy clearance in this chapter) assessment. Indeed the risk and privacy clearance are pre-computed and can be extracted respectively from the Policy repository and Attributes repository (represented in Figure 6.1). Of course, these modules can be integrated into our architecture if we will include dynamic risk and privacy clearance assessment in a modified version of this differential privacy based model as we will discuss later in this chapter.

6.6 Experimental Evaluation

In order to evaluate the practical feasibility of our approach, we developed a proof-of-concept implementation of the framework, to assess: *i)* the impact of our privacy-preserving access control on the data quality. To this aim, we defined a simple classification task, and test the performance using data sanitized at different privacy clearances. *ii)*

to evaluate the impact of the enforcement of different privacy clearance levels (anonymization by applying differential privacy) on the performance of our access control system, in terms of response time.

To address these questions, we implemented a prototype of our Privacy Enforcement Module as described in Section. 6.5. As data sanitizer we used *"DiffGen"* a Differentially-private anonymization algorithm based on Generalization, proposed and implemented by Mohammed et *al.* in [113].

DiffGen anonymizes the raw data by probabilistically generalizing the attributes. More in details, starting from the most general state (one-single group), a set of specializations are randomly selected, using an exponential mechanism with a predefined scoring function (e.g. a utility-based function assessing the information gain for each specialization). Then, the algorithm computes the contingency tables, counting the number of records sharing a combination of attributes, and, it applies Laplacian noise, with variance ϵ, to the generalized contingency table. Synthetic data can be then produced from the generalized and randomized contingency table (see in [113] for details). The resulting data set, generated by a ϵ-differential privacy mechanisms, can be safely used for any data analysis.

In Table 6.3 we represent different scenarios introduced in our use case in Section. 6.2. We also map each scenario with the query's risk level and required privacy clearance $\mathcal{T}$ [6].

Table 6.3: Example of risk and privacy clearance levels for different access scenarios introduced in the use case Section. 6.2.

#	**Role**	**Operation**	**Risk level**	**Required (clearance)**
1	HR manager	HR view (int.)	Low Risk	$\mathcal{T}_2 = [0.01, 1[$
2	HR manager	HR view (ext.)	Medium Risk	$\mathcal{T}_3 = [1, 10[$
3	HR developer	Testing data	High Risk	$\mathcal{T}_4 = [10, 20[$
4	HR Benchmarking	Benchmark	Very-High Risk	$\mathcal{T}_5 = [20, +\infty[$

For our test, we use the Adult Data Set [7] from the UCI Machine Learning Repository. This dataset contains $45K$ records from the US Census dataset with 15 demographic and employment-related variables (6 numerical, 8 categorical, and 1 binary class column representing two income levels, $\leq 50K$ or $> 50K$). The Experiments were conducted on an Intel Core i5 2.6GHz PC with 8GB RAM.

Data Quality To evaluate the impact of anonymization on the Utility of data we propose to assess its impact on the accuracy of a simple classifier trained and tested using anonymized data at different clearance levels (shown in Table 6.3).

We use as (binary) class attribute the income level, $\leq 50K$ or $> 50K$, and as classifier

[6] the full mapping between the risk levels and required privacy clearances can be found in Table 6.2

[7] Available at http://archive.ics.uci.edu/ml/datasets/Adult

the well-known C4.5 Algorithm [133]. Each anonymized data set is split in two. First part of the data (2/3) is used as training data to build a classifier, and the remaining data (1/3) is used as test data to measure the classification accuracy.

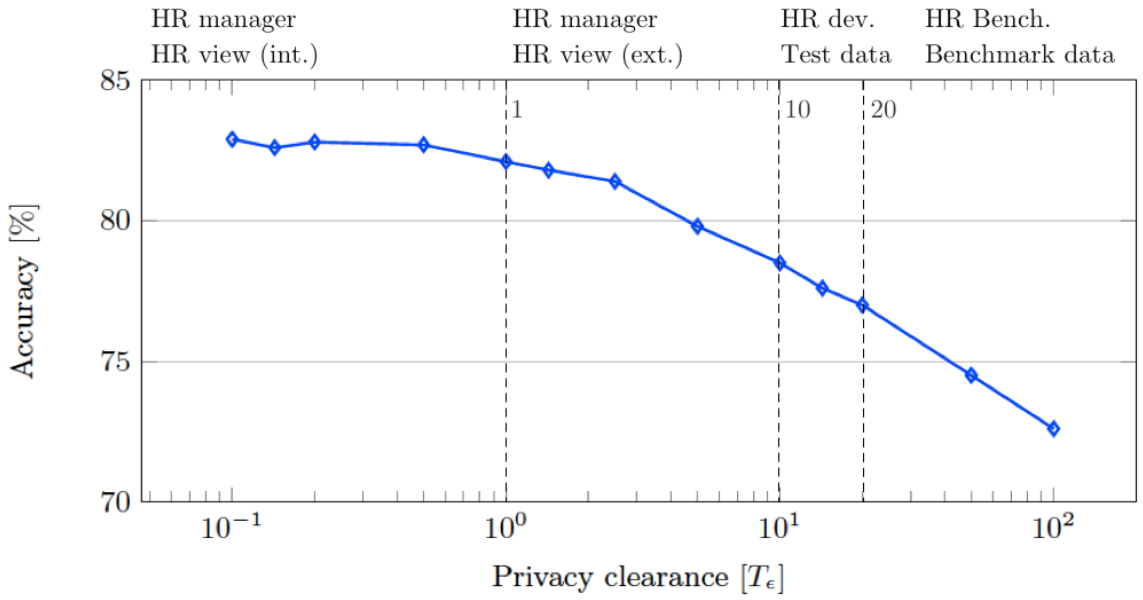


Figure 6.2: Classifier Accuracy for different privacy clearance T_ϵ in different intervals $\mathcal{T}$. Each data point represent the average over 100 runs (parameters of DiffGen: number of specialization of specialization $N_s = 10$, and scoring function $u = Max$).

In Figure. 6.2, we report the accuracy of classifiers for different privacy clearances. We can observe that for queries with *Very-High Risk* risk level, requiring a large privacy clearance $T_\epsilon \in \mathcal{T}_5$ (where we sanitize the data to obtain small values of $\epsilon = 1/T_\epsilon$), the accuracy is highly impacted.

In fact, with $\epsilon = 0.01$, the attributes are almost fully generalized, and the accuracy is close to the case where all the attributes (but the class attribute, of course) are removed. Still, the accuracy level of $\simeq 75\%$ could be enough for many benchmarking tasks.

The accuracy goes up, as expected, for when lower privacy clearance values are required. Privacy clearance in the range $\mathcal{T}_4 = [10, 20[$, still considered reasonably safe in practical cases, allows producing data able to provide an accuracy close to 80%, which it could be sufficient as testing data for development, and to have a general view for a manager on other department analytics.

Privacy clearance > 1 (manager view on own team data, in our use case), gives levels of accuracy close to the raw data $\simeq 85\%$.

Performance We estimate the computational overhead caused by the sanitization. From these experiments, we observe that the time for performing the sanitization can be easily

of the order of seconds, see Figure. 6.3. The effect of the required privacy clearance (and the *epsilon* for the required level of sanitization) on the performance (time) is limited.

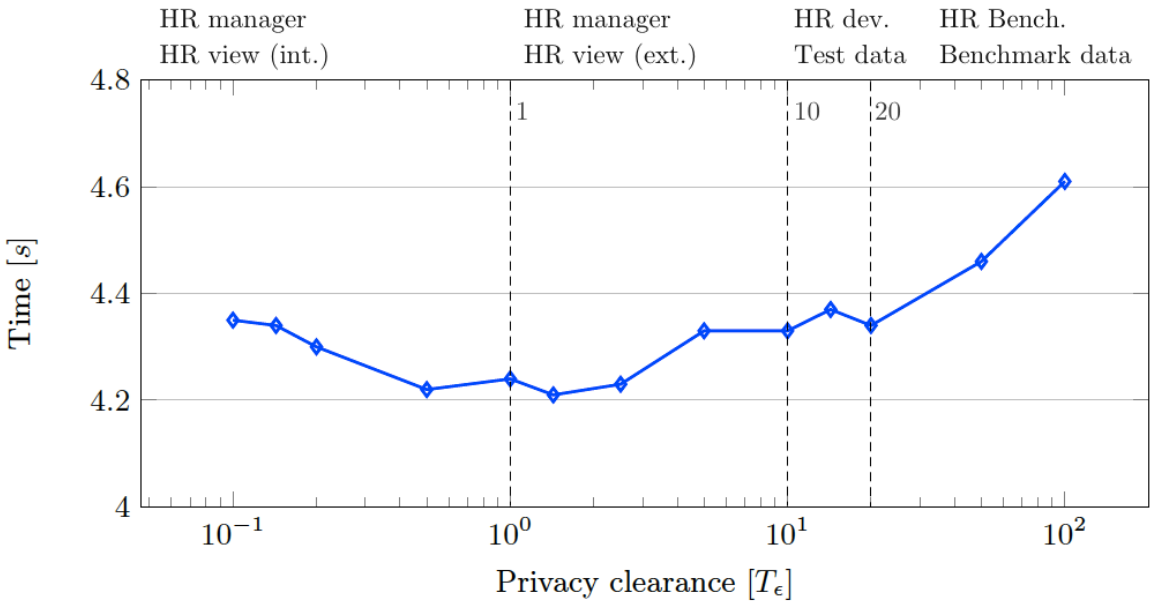


Figure 6.3: Anonymization time for different privacy clearance T_ϵ. Each data point represent the average over 100 runs (parameters of DiffGen: number of specialization of specialization $N_s = 10$, and scoring function $u = Max$).

Despite being preliminary results, it is clear that for reaching real-time performance (as it is possible for k-anonymity algorithms, see Chapter 4), it is needed to include some optimization, for example in terms of caching or testing other algorithms for generating differential private data set.

6.7 Chapter conclusions

In this chapter, we proposed a novel privacy-aware access control model, based on differential privacy. The model allows for data access at different privacy levels, generating a sanitized data set according to the privacy clearance of the request. This model is complementary to the one based on syntactic anonymity (to which we devoted the previous chapter), indeed although differential privacy was originally devised for privacy-preserving data mining (so to answer a limited number of specific queries), it is becoming popular also for privacy-preserving data publishing (so to produce anonymized dataset) [167], as in our case. As we have shown, here, using a risk-based access control model in combination with differential privacy, we are able to provide a flexible access control mechanism producing sufficiently accurate data for a classification task, and, at the same time, with

a formal guarantee of the privacy level.

To evaluate our approach we developed a proof-of-concept prototype. A first experimental analysis, considering an HR related use case, and a benchmarking dataset, indicates that the model can address complex privacy and utility requirements. Indeed, in our use case, we propose to integrate differential privacy within a privacy-aware risk-based access control model to prevent classification model from violating the privacy of individuals in the training data while ensuring a decent level of accuracy to allow different actors to exploit the results of this analysis.

This approach still presents a number of open issues to be solved for a practical usage. For example, the performance of the current implementation is not a real-time performance, therefore different algorithms and optimization strategies for the anonymization need to be investigated.In addition, whereas in previous models we used the concept of privacy risk, which has a clear business interpretation, here we used the ϵ parameter of differential privacy. In future works, we would like to relate the two approaches, including explicitly privacy-risk assessment and adjustment mechanisms based on the concepts of differential identifiability [90] and interactive differential privacy [54].

Chapter 7

Evaluation of the Privacy-aware Risk-based access control model Using EPIC

In this chapter, we propose to evaluate our privacy-aware risk-based access control approach from privacy enhancement perspective using the EPIC methodology. We apply EPIC to identify and evaluate privacy threats originated by authorized insider actors for two cybersecurity systems (CSS): a) a "classic TDS" and b) the same TDS equipped with our privacy-aware access control "privacy-aware TDS". Then comparing the risk level of insider privacy threats (i.e., privacy threats originated by authorized insider actors) identified in both CSS.

The results of this evaluation show that the privacy violation risk of privacy threats for several actors is significantly mitigated after the implementation of our privacy-aware risk-based access control system.

7.1 Introduction

Threat Detection Systems TDS (described in Chapter 5, Section 5.5) are CSS (cybersecurity systems) used by organizations to monitor their information systems and detect security threats and anomalies. A TDS collects and processes security information from several entities deployed in the organization's network (e.g., routers, end-user machines, other CSS). Human agents (security experts) constantly interact with the TDS to accomplish several tasks e.g., monitoring security events, investigating alerts, maintaining improving the threat detection. The data collected by TDS and accessible by the agents often contain private information about individuals in the organization (e.g., employees,

collaborators, clients). Therefore a TDS can be the source of several privacy violation threats that the organization should identify, evaluate, and mitigate. These privacy violation threats include threats originated the security agents, who are authorized to access data but can abuse their access privileges, not respect the purposes of accessing data, and violate privacy.

In Chapter 5, we propose a privacy-aware risk-based access control system as a solution to mitigate this category of threats. In this chapter, we evaluate our privacy-aware risk-based access control system from the privacy perspective and highlight its mitigatory impact on threats identified for a TDS deployed in a testing environment (as described in Section 5.5).

To identify the privacy threats, and evaluate the impact of our solution on these threats, we will use the methodology EPIC (described in Section 4)). As a reminder, EPIC (Evaluating privacy risk in cybersecurity systems) is a four-step methodology used to identify, evaluate and prioritize privacy violation threats in CSS. In addition, EPIC can also be used to compare two CSS from the privacy perspective or a CSS before and after the adoption of a privacy-enhancing feature. Consequently, the methodology can also be used to evaluate a given privacy-enhancing solution by assessing the privacy improvement brought by the adoption of this solution.

Since the evaluation carried in this chapter aims at studying the impact of our privacy-aware risk-based access control on the privacy threats originated by authorized actors, we will only focus on the aspects of the TDS related to these actors and components equipped with the access control system. Moreover there will be no need to go through the threat prioritization (last part of EPIC Step 4 Section 3.3.4), since we are not expecting the privacy-aware risk-based access control to have an impact threat priority level (from adversaries trust perspective), we are rather interested in observing its mitigatory impact on the risk level.

The remaining of this chapter will be structured as follows: In Section 7.2, we will apply the EPIC analysis to the Threats Detection System “TDS” (described in Section 5.5). in Section7.3 e will apply the EPIC analysis to the “Privacy-aware TDS” and highlight the we will summarize, comment the results and conclude in Section 7.4

7.2 Privacy evaluation: TDS

In this section, we report the main results of privacy threat analysis of the TDS. The privacy threat analysis is achieved through the application of the evaluation methodology EPIC presented in chapter 3.

7.2.1 EPIC Step 1 (TDS)

The aim of this first step "*Model the cybersecurity system*" (described in 3.2.2) is to understand the cybersecurity system architecture, its data flows, and its functional aspects. For this scope, we model the TDS using an extended data-flow diagram DFD+ (the different elements of DFD are described in Figure 3.2 . The elements added by the extended version DFD+ are described in Figure 3.3).

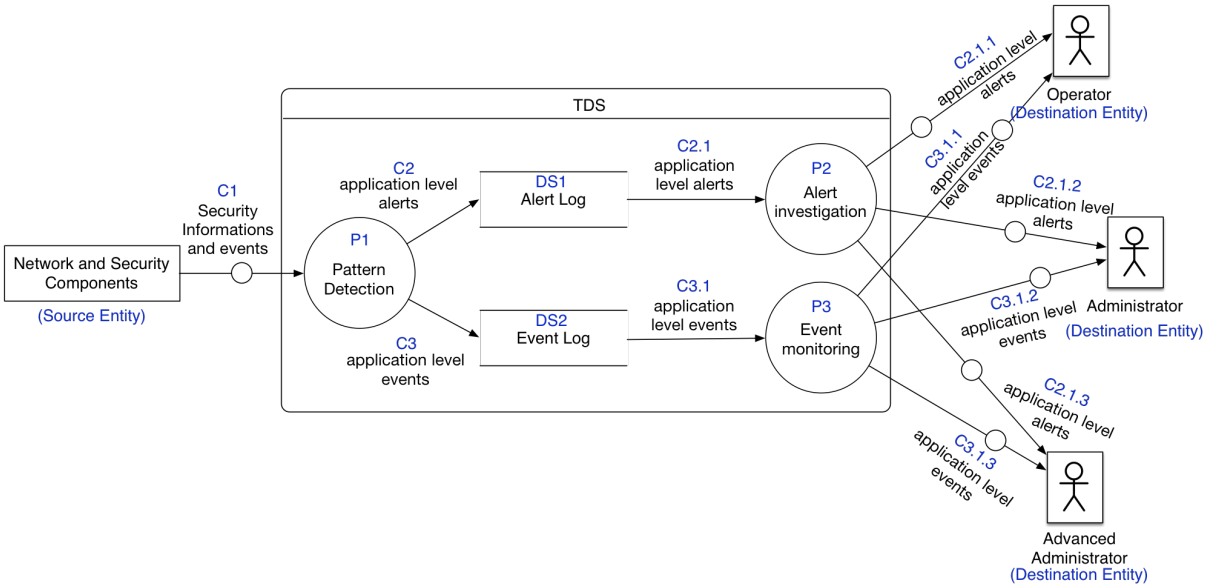


Figure 7.1: DFD+ model (TDS)

Figure 7.1 depicts different components and data flows in the TDS. Security information and events are constantly collected by different security (e.g., intrusion detection, firewalls) and networking (e.g., routers, servers) components, deployed in the information system. This information is sent to the TDS through the channel *C1* and it goes through a pattern matching executed by the process *P2* to detect attacks and malfunctioning alerts (that needs to be further investigated) as well as suspicious events (that needs to be monitored to prevent other attacks and anomalies). After transiting through the channels *C2* and *C3*, alerts and events are respectively stored in the storage *DS1* and *DS2*. The logs can be retrieved from the data storage through channels *C2.1* to the process *P2* and *C3.1* to *P3* The process *P3* enables the three actors (i.e., *Operator*, *Administrator* and *Advanced administrator*) to investigate the alerts logged by the TDS. Data is delivered to the actors through *C2.1.1*, *C2.1.2* and *C2.1.3*. The process *P3* allows the actors to monitor suspicious events delivered through *C3.1.1*, *C3.1.2* and *C3.1.3*.

7.2.2 EPIC Step 2 (TDS)

The aim of the second step "*Identify data exposure*" is to identify and evaluate possible data exposures, i.e., situations in which data is disclosed to a potential adversary (see Section 3.2.3 for more details about this step).

Table 7.1: Adversaries table (TDS)

Adversary	**Description**
Operator	Classify alerts and report patterns anomalies His/Her tasks require access to pattern detection results (events/log data related to the suspicious pattern) in case of alerts.
Administrator	Has all *Operator* tasks and privileges. They can also Investigate alerts, Create or Reconfigure patterns. He/She should have access the detection results and events data related to the patterns.
Advanced administrator	Has all *Administrator* tasks and privileges. Can also grant exceptional access to the data by attributing higher trust level to an *Operator* or an *Administrator*.

As explained in the introduction we will mostly focus on aspects of the analysis related to authorized actors. We first identify different actors/adversaries interacting with the TDS. In Table 7.1 we report a list of authorized adversaries (these adversaries have been first introduced in Section 5.5). After that, we study different security mechanisms protecting of components identified in step 1 (depicted in Figure 7.1). Actors, in our case, have authorized interacts only with processes *P2* and *P3* (these interactions are controlled by an access control system). In Table 7.2 we report the security mechanisms protecting these two processes.

Table 7.2: Components security table (TDS)

Component	**Authorized users**	**Security mechanisms**
P2	Advanced admin., Administrator, Operator	Access control, authentication, network security
P3	*same as above*	Access control, authentication, network security

Once we have the description of adversaries and components, we can now establish the list of exposures and assess their magnitudes and the likelihoods of access. We report the results of this assessment in Table 7.3.

We identify three exposures for each for each component: $Exp1$, $Exp2$, and $Exp3$ at the level of process *P2* and $Exp4$, $Exp5$, and $Exp6$ at the level of process *P3*. As mentioned earlier, actors are authorized to access the data from processes *P2* and *P3* consequently the likelihood of access $L_a = Authorised$ for all six exposures in Table 7.3. The magnitude of exposure in a process is assessed considering the number and frequency of access for each actor and the amount of data available to access i.e., amount of data collected

Table 7.3: Data exposures table (TDS)

Component	**Adversary**	**Exposure**	L_a	**Exposure magnitude**
P2: Alert investigation	Advanced admin.	$Exp1$	Authorized	Important
	Administrator	$ExP3$	Authorized	Important
	Operator	$Exp3$	Authorized	Important
P3: Events monitoring	Advanced admin.	$Exp4$	Authorized	Very-Important
	Administrator	$Exp5$	Authorized	Very-Important
	Operator	$Exp6$	Authorized	Very-Important

daily, retention time (see Section 3.2.4 for more details). All our actors in our case have unlimited access the data both for alert investigation (*P2*) and event monitoring (*P3*). In addition, our threat detection system offers the possibility to collect a huge amount of data which companies store for an extended amount of time. For example, our data security events data set has over $1bn$ entry collected in a testing environment in a period shorter then a month this amount can be way bigger in production environment e.g., an organization with 1000 employees will process in average 3.5 terabytes of security data monthly [65].

7.2.3 EPIC Step 3 (TDS)

After identifying the exposures in the previous steps, we now assess whether these exposures (summarized in Table 7.3) represent privacy threats or not (see Section 3.2.4 for mode details about Step 3 "*Identify privacy threats*"). To do so we need to take into account which attributes and which types of data are actually exposed, thus we start by listing and describing these data attributes.

For example, the TDS we have been studying (SAP ETD) logs and uses over 40 attributes related to security events. However, not all of these attributes are interesting for us (e.g., meta-data of filters and patterns, processing time stamps, log type), therefore we dismissed these attributes during the analysis. Some of these attributes are also privacy neutral (i.e., not identifiers, not QIDs, nor sensitive), and, for the sake of brevity, we do not mention these attributes either during the analysis. In Table 7.4 we report a sample of relevant attributes, and we provide a description and an example of each attribute.

In the example given in Section 3.2.3, we explained the data leaked in each exposure is composed of heterogeneous types of records (i.e., records with different attributes). Our TDS, however, can correlate different events from several types of logs (and other sources) to generates unified logs that are as homogeneous and rich as possible. Nonetheless, the data views exposed through processes *P2* and *P3* have different structure (i.e., different attributes) depending on the pattern (anomalies and security threat patterns described in Section 5.5) generating the alerts to investigate or the events to monitor. Therefore,

Table 7.4: Attributes description table (TDS)

Name	Description	Domain	Example
Timestamp	Event (log entry) timestamp	DD.MM.YYYY hh:mm:ss	07.10.2015 18:39:36
UserID	pseudo-anonymised user identifier	Alphanumeric pseudo	DG70W98CY1
SysID (origin)	Identifier of the system originating a request (client or server)	Alphanumeric code	MAIL/X009
SysID (target)	Identifier of the system targeted a request (client or server)	Alphanumeric code	ERP/E113
Hostname	Machin ID (often organized by geolocation)	Alphanumeric code	ITA-Trento-ND0606
IP (origin)	IP adress (source) of a machine in the internal or external network	IP Address	91.218.36.178
IP (target)	IP adress (destination) of a machine in the internal or external network	IP Address	239.121.10.177
PrivilegeName	Name describing a given privilege (action, transaction)	action-TransactionName	create-FS-45 (create an outgoing payment)
TransactionName	Technical name of a specific business transaction	Alphanumeric code	FS-45 - outgoing payments
File(metadata)	File name, download path, size, author creation time etc.		path\name.pdf, 504kb, 2017-06-06 12:07:10
File(content)	A file being downloaded	String of bytes	
Email(header)	Email Object, Sender and Reciver adresses	smtp header	from: to: date: subject: etc.
Email(content)	Content of an Email(textual) and attachments		
URL (path)	visited sites urls and parameters if any	URL	https://youtube.com/watch?=mmtgs

in our analysis, different data contents will be defined for each used pattern. For sake of brevet, we will only report some of the most commonly used patterns.

In Table 7.5[1] we describe the patterns that we will be used and the data attributes collected and (can be exposed) by each pattern. A data content will be associated to each pattern or more exactly to the data exposed by a pattern (e.g., $dc1$ will be associated to *Brute force attack* pattern and will expose the following attributes *IP (origin)*, *IP (target)*, *UserID*, *Hostname*, and *Timestamp*).

As required by the EPIC methodology, we continue our analysis by classifying the attributes of each data content as identifying (ID), quasi-identifying (QID), or potentially

[1]Note that this Table 7.5 does not belong to the list of tables used for EPIC (see Figure 3.1). We use this table in this use case for clarity proposes.

Table 7.5: Data contents patterns association, description of the patterns, and attributes used by each pattern (TDS)

Data content	**Pattern Name**	**Pattern description**	**Attributes collected**
*dc*1	Brute force attack	***Event***: failed login attempt ***Alert***: 20 failed login attempts in less than 10 minutes	IP (origin); IP (target); UserID; Hostname; Timestamp
*dc*2	Irregular transactions	***Event***: failing transaction ***Alert***: successful transaction after failure or Blacklisted transaction	IP (origin); IP (target); UserID; Hostname; SystemID (target); TransactionName; PrivilegeName; Timestamp
*dc*3	Blacklisted URL	***Alert*** (and ***Event***): request to access a blacklisted URL	IP (origin); IP (target); UserID; Hostname; SystemID (origin); Timestamp; URL
*dc*4	Multiple downloads by one user (Files)	***Event***: a file is downloaded to a monitored system ***Alert***: file size or files number exceeds the allowed threshold	IP (origin); IP (target); UserID; Hostname; SystemID (origin); File(metadata); File(content); Timestamp
*dc*5	Multiple downloads by one user (Emails)	***Alert*** (and ***Event***): email size or number exceeds the allowed threshold	IP (origin); IP (target); UserID; Hostname; SystemID (origin); Email(header); Email(content); Timestamp

sensitive information (PSI). When an attribute is classified as QID we indicate which background knowledge may lead to re-identification when joined with the attribute value. Results are reported in Table 7.6.

Finally, in Table 7.7 we show which data contents are exposed by each component. 5 data contents have been identified for of the processes *P2* and *P3* since the two processes mostly use the same patterns (in our use case), the data contents exposed are similar for both of them. In addition all adversaries (*Advanced administrator* ,*Administrator*, and *Operator*) have access to all data contents. In this step, no privacy threat (i.e., a combination of exposure and data content) will be cleared and all identified threats will be further assessed in the next section.

7.2.4 EPIC Step 4 (TDS)

EPIC's fourth step "*Evaluate and prioritize privacy threat risk*" aims at evaluating and prioritizing the identified privacy threats (see Section 3.3 for more details about this step). In this chapter, however, we will omit the very last part of Step 4 "the threat prioritization" since we are only interested in comparing the privacy risk before and after adopting the privacy-aware risk-based access control model as a privacy risk mitigation solution.

As defined by EPIC, we start this step by evaluating privacy violation likelihood of

Table 7.6: Data content attributes analysis table (TDS)

Data content	**ID**	**QID**		**PSI**
		Attribute	**Bg. Kowledge**	
*dc*1	None	IP (origin); UserID; Hostname	List associating IP-addresses or UserIDs or Hostnames with user-names	IP (target); Timestamp
*dc*2	None	IP (origin); UserID; Hostname; PrivilegeName	List associating IP-addresses or UserIDs or Hostnames with user-names. Knowledge about privilege-users assignment (especially for very particular privileges)	IP (target); SystemID (target); TransactionName; Timestamp
*dc*3	None	IP (origin); UserID; Hostname; SystemID (origin) URL	List associating IP-addresses or UserIDs or Hostnames with user-names	IP (target); Timestamp; URL
*dc*4	None	IP (origin); UserID; Hostname; SystemID (origin) File(metadata); File(content)	List associating IP-addresses or UserIDs or Hostnames with user-names	IP (target); File(metadata); File(content)
*dc*5	None	IP (origin); UserID; Hostname; SystemID (origin); Email(header); Email(content)	List associating IP-addresses or UserIDs or Hostnames with user-names	IP (target); Email(header); Email(content)

Table 7.7: Data content identification table (TDS)

Exposure				**Data content**
Exp.	**Component**	**Adversary**	L_a	
*Exp*1	P2. Alert Investigation	Advanced administrator	Authorized	*dc*1, *dc*2, *dc*3, *dc*4, and *dc*5
*Exp*2	P2.	Administrator	Authorized	*Same data contents as above*
*Exp*3	P2.	Operator	Authorized	*Same data contents as above*
*Exp*4	P3. Event Monitoring	Advanced administrator	Authorized	*dc*1, *dc*2, *dc*3, *dc*4, and *dc*5
*Exp*5	P3.	Administrator	Authorized	*Same data contents as above*
*Exp*6	P3.	Operator	Authorized	*Same data contents as above*

privacy violation threats (see Table 7.8), then impact severity of these violations (see Table 7.9), and finally we evaluate the privacy violation risk (Table 7.10) as the combination of privacy violation likelihood and impact severity using the risk matrix presented in Section 3.3.3 in Table 3.10. In the following, we provide some details on how these values were obtained.

Table 7.8: Privacy violation likelihood table (TDS)

Exposure	**Data content**	L_a	L_{rid}	Th	L
$Exp1$: P2 Advanced admin.	$dc3$	Authorized	Medium	$Th1$	**High**
	$dc2$	Authorized	High	$Th2$	**Very High**
	$dc3$	Authorized	High	$Th3$	**Very High**
$Exp2$: P2 Administrator	$dc4$	Authorized	High	$Th9$	**Very High**
	$dc5$	Authorized	Certain	$Th10$	**Very High**
$Exp3$: P2 Operator	$dc4$	Authorized	High	$Th14$	**Very High**
	$dc5$	Authorized	Certain	$Th15$	**Very High**
$Exp1$: P3 Advanced admin.	$dc3$	Authorized	Medium	$Th16$	**High**
	$dc2$	Authorized	High	$Th17$	**Very High**
	$dc3$	Authorized	High	$Th18$	**Very High**
$Exp2$: P3 Administrator	$dc4$	Authorized	High	$Th24$	**Very High**
	$dc5$	Authorized	Certain	$Th25$	**Very High**
$Exp3$: P3 Operator	$dc4$	Authorized	High	$Th29$	**Very High**
	$dc5$	Authorized	Certain	$Th30$	**Very High**

In threats $Th1$, $Th2$, and $Th3$ data contents $dc1$, $dc2$, and $dc3$ are (respectively) exposed (in $Exp1$) at the level of process $P2$ (*Alert investigation*) to the adversary *Advanced administrator* ($Exp1$) who has a likelihood of access $L_a = Authorized$ (see Table 7.8). In $Th1$, the data content $dc1$ contains (among others) the QID attribut *IP(origin)* that can be used by the adversary to reidentify the records with a *medium* likelihood ($L_{rid} = medium$). Indeed the *Alert investigation* is not supposed to access information mapping the *IP(origin)* and the identity of respondents. He/She can however easily gain this information a long exercising his/her security tasks. $Th2$ and $Th3$ have a re-identification likelihood $L_{rid} = High$. In addition to the QID *IP(origin)* data content $dc2$ and $dc3$ (exposed by $Th2$ and $Th3$) contain the QID attributes *PrivilegeName* and *SystemID (origin)* (see Table 7.6) which can increase the likelihood of re-identification especially since the *Alert investigation* has a fairly high capacity of obtaining this information especially for particular Privileges (e.g., high-level privileges often assigned to very few people) and Systems(e.g., very old or very new systems are easy to single out).

The impact of $Th1$ and $Th2$ is *low* (see Table7.9). The violation magnitude of these threats is quite limited (respectively *limited* and *very-limited*) because there are very few alerts related to $dc1$ and $dc2$ (they are mostly collected as events) in addition data attributes in these two data contents are not very sensitive (mostly internal IP addresses and work related information) therefore in the context of a security alert both non-compliance impact I_C and reputation Loss impact I_R[2] have low levels. $Th3$ however has an *important*

[2]In this evaluation, we do not have data to evaluate the impact of privacy violation in terms of business agreements (failure to meet business agreements). Therefore, and for sake of simplicity, we will assume the company has no privacy business agreements to comply with, and we will not consider the impact factor I_B

Table 7.9: Qualitative privacy violation impact table (TDS)

Exposure	**Data content**	**Th**	**Violation magn.**	I_C	I_R	I
$Exp1$: P2 Advanced admin.	$dc1$	$Th1$	Limited	Low	Low	**Low**
	$dc2$	$Th2$	Very-limited	Low	Low	**Low**
	$dc3$	$Th3$	Important	Low	Medium	**Medium**
$Exp2$: P2 Admin.	$dc4$	$Th9$	Medium	Med-high	Med-high	**Med-high**
	$dc5$	$Th10$	Limited	Med-high	Med-high	**Med-high**
$Exp3$: P2 Operator	$dc4$	$Th14$	Medium	High	High	**High**
	$dc5$	$Th15$	Limited	High	High	**High**
$Exp4$: P3 Advanced admin.	$dc1$	$Th16$	Medium	Low	Low	**Low**
	$dc2$	$Th17$	Medium	Low	Med-low	**Med-low**
	$dc3$	$Th18$	Important	Low	Medium	**Medium**
$Exp5$: P3 Admin.	$dc4$	$Th24$	Medium	High	High	**High**
	$dc5$	$Th25$	Limited	High	High	**High**
$Exp6$: P3 Operator	$dc4$	$Th29$	Medium	High	High	**High**
	$dc5$	$Th30$	Limited	High	High	**High**

magnitude of violation and $dc3$ contains URLs which might leak important information about respondents. This reflects on $Th3$ risk level $R = high$ (see Table 7.17).

In threats $Th16$, $Th17$, and $Th18$ the same data contents $dc1$, $dc2$, and $dc3$ are exposed (in $Exp4$) to the same adversary (*Alert investigation*) in an other context (event monitoring) from process *P3*. Overall these threats slightly higher risk (with respect to $Th1$, $Th2$, and $Th3$. see Table 7.17). In fact, the magnitude of violation in this context (Event monitoring) is more important for all three threats because generally more data is collected as events (the number of events is higher than the number of alerts) and this affects more respondents. In addition access to some sensitive data might not be always justifiable (in the public opinion) if there is no real need (security alerts) and if this access is used to violate privacy the impact on reputation can be important (e.g., in the case of $Th18$). (e.g., in the case of $Th18$).

$Th9$, $Th10$, $Th14$ and $Th15$ describe the threats of exposing data contents $dc4$ (in $Th9$ and $Th14$) and $dc5$ ($Th10$ and $Th15$) by the process *P2* to two different actors (*Administrator* and *Operator*). The above threats, all, have very high likelihoods of re-identification $L_{rid} = high$ or $L_{rid} = certain$ (see Table 7.8) which overall leads to very high likelihoods of violation (L). This is due to the fact that both data contents exposed in these threats ($dc4$ and $dc5$) leak very strong QIDs (highly identifying attributes/attribute combinations e.g., *Email(header)*, *Files(meta)* and *Files(content)* see Table 7.6). These threats have average magnitude of violation (*limited* to *medium*) because, although very sensitive $dc4$ and $dc5$ are not collected very .

$Th24$, $Th25$, $Th29$, and $Th30$ are threats identified at the level of *P3*. They expose data

Table 7.10: Qualitative privacy violation risk table (TDS)

Th	**Exposure**	**Data content**	L	$\mathcal{I}$	R
$Th1$	$Exp1$: P2 Advanced admin.	$dc1$	High	Low	**Low**
$Th2$	$Exp1$: P2 Advanced admin.	$dc2$	Very-high	Low	**Medium**
$Th3$	$Exp1$: P2 Advanced admin.	$dc3$	Very-high	Medium	**High**
$Th9$	$Exp2$: P2, Administrator	$dc4$	Very-high	Med-high	**High**
$Th10$	$Exp2$: P2, Administrator	$dc5$	Very-high	Med-high	**High**
$Th14$	$Exp3$: P2, Operator	$dc4$	Very-high	High	**High**
$Th15$	$Exp3$: P2, Operator	$dc5$	Very-high	High	**High**
$Th16$	$Exp4$: P3 Advanced admin.	$dc1$	High	Low	**Low**
$Th17$	$Exp4$: P3 Advanced admin.	$dc2$	Very-high	Med-Low	**Medium**
$Th18$	$Exp4$: P3 Advanced admin.	$dc3$	Very-high	Medium	**High**
$Th24$	$Exp5$: P3, administrator	$dc4$	Very-high	High	**High**
$Th25$	$Exp5$: P3, administrator	$dc5$	Very-high	High	**High**
$Th29$	$Exp6$: P3, Operator	$dc4$	Very-high	High	**High**
$Th30$	$Exp6$: P3, Operator	$dc5$	Very-high	High	**High**

contents $dc4$ and $dc5$ to the actors *Administrator* and *Operator* in the context of event monitoring. Similarly to $Th9$ and $Th10$, $Th24$ and $Th25$ have very high likelihoods of violation (L see Table 7.8) since they expose the same data contents ($dc4$ and $dc5$). Despite also having the same magnitude of violation, $Th24$ and $Th25$ have higher overall impact severity (I see Table 7.9) for when involving the actor *Administrator* (with respect to $Th9$ and $Th10$). This increase of the impact severity is due to the lack of appropriateness in the decision of granting these two actors (with average levels of trust) access to this very sensitive data (in $dc4$ and $dc5$) without a real need for this leakage, both from compliance and reputational standpoint. We can observe the same trends with threats $Th29$ and $Th30$ involving the actor *Operator* which have similar privacy violation likelihood that $Th14$ and $Th15$ (threats involving the same actor and exposing the same data contents $dc4$ and $dc5$ in a different context). The impact severity level changes were not very observable in this case because the impact levels are already at their highest level see Table 7.9, however, a quantitative assessment would probably reveal slightly higher values.

7.3 Privacy evaluation: Privacy-aware TDS

In this section, we apply the EPIC methodology on a TDS equipped with the privacy-aware risk-based access control system (we refer to this TDS as "Privacy-aware TDS"). We will then compare these results with the results obtained in the last section (reporting the EPIC analysis of the TDS without privacy-aware risk-based access control system "classic TDS") to assess the privacy improvements.

7.3.1 EPIC Step 1 (Privacy-aware TDS)

The architecture of the TDS (components, and actors interacting with the system) is not affected by the adoption of the privacy-aware risk-based access control system. Therefore we will use the DFD+ diagram modeling depicted in Figure 7.1) for the rest of the analysis.

7.3.2 EPIC Step 2 (Privacy-aware TDS)

The adversaries interacting with the Privacy-aware TDS are the same adversaries identified in Table 7.1, they also have the same description and tasks.

Table 7.11: Components security table (Privacy-aware TDS)

Component	**Authorized users**	**Security mechanisms**
P2	Operator, Administrator, Advanced Admin.	Privacy-aware risk-based access control, authentication, network security
P3	*same as above*	*same as above*

The security mechanisms implemented to protect processes *P2* and *P3*[3] of the privacy-aware TDS (reported in Table7.11) are similar to the mechanisms identified for the classic TDS (reported in the previous section Table7.2) with the exception, of course, of the privacy-aware risk-based access control equipping the privacy-aware TDS.

Table 7.12: Data exposures table (TDS)

Component	**Adversary**	**Exposure**	L_a	**Exposure magnitude**
P2: Alert investigation	Advanced admin.	$Exp1$	Authorized	Important
	Administrator	$Exp2$	Authorized	Important
	Operator	$Exp3$	Authorized	Medium
P3: Events monitoring	Advanced admin.	$Exp4$	Authorized	Very-Important
	Administrator	$Exp5$	Authorized	Important
	Operator	$Exp6$	Authorized	Limited

The adoption of the privacy-aware risk-based access control (in the Privacy-aware TDS) does not affect the likelihood of access (L_a) of the identified exposures, as reported in Table 7.12 (with respect to the one reported in Table 7.3). The magnitude of exposure, however, will slightly decrease especially in *P3* and for the adversaries *Administrator* and *Operator*, since this process (Events monitoring) does not have the same emergency level than *P2* (Alert investigation), in addition, it can be carried on using partially obfuscated data. If needed the *Administrator* and *Operator*, according to their trust levels, can request additional data in return for fulfilling trust enhancement obligations. The *Advanced*

[3]Similarly to the previous section, Section 7.2 we are only interested in studying processes *P2* and *P3* since they are the only component with which authorized actors interact through an access control system

administrator is expected to have a very high trust level, and his ability to access even very risky data will be very little affected by the new access control system.

7.3.3 EPIC Step 3 (Privacy-aware TDS)

After identifying the exposures in the previous steps, we now assess whether these exposures (summarized in Table 7.3) represent privacy threats or not (see Section 3.2.4 for mode details about Step 3 "*Identify privacy threats*").

Table 7.13: Data contents patterns association, description of the patterns, and attributes used by each pattern (Privacy-aware TDS)

Data content	**Pattern Name**	**Pattern description**	**Attributes collected**
*dc*4.1	Multiple downloads by one user (Files)	***Alert***: file size or files number exceeds the allowed threshold	IP (origin); IP (target); UserID; Hostname; SystemID (origin); File(metadata); File(content); Timestamp
*dc*4.2	Multiple downloads by one user (Files)	***Event***: a file is downloaded to a monitored system	IP (origin); IP (target); UserID; Hostname; SystemID (origin); File(metadata); Timestamp
*dc*5.1	Multiple downloads by one user (Emails)	***Alert***: email size or number exceeds the allowed threshold	IP (origin); IP (target); UserID; Hostname; SystemID (origin); Email(header); Email(content); Timestamp
*dc*5.2	Multiple downloads by one user (Emails)	***Event***: email size or number exceeds the allowed threshold	IP (origin); IP (target); UserID; Hostname; SystemID (origin); Email(header); Timestamp

The attributes exposed by the privacy-aware TDS are the same attributes exposed by the classic TDS. A list of the most relevant attributes is reported and described Table 7.4.

In addition, the new access control system introduces *dc*4.1, *dc*4.2, *dc*5.1 and *dc*5.2. These are sub-contents respectively of *dc*4 and *dc*5 (see Table 7.5). *dc*4.2 and *dc*5.2 are sub-contents from which we remove the most sensitive attributes *Email(content)* and *File (content)*. *dc*4.1 and *dc*5.1 have exactly the same attributes than *dc*4 and *dc*5. The access control system, now, distinguishes between data contents, released for event monitoring (i.e., *dc*4.2, *dc*5.2), and data contents released for alert investigation (i.e., *dc*4.1 and *dc*5.1) for the attack patterns *"Multiple downloads by one user (Emails)"* and *"Multiple downloads by one user (files)"*(see Table 7.13). Indeed, the actors (see Table 7.1) do not need to see the *Email(content)* and *File (content)* when monitoring events for these two patterns, so there is no need to take a risk in this context, especially when the request does not have high-level of trust (e.g., low trust user, low trust device).

The data contents exposed by each exposure is also slightly modified by the adoption of the privacy-aware risk-based access control systems for some of the actors, as reported in Table 7.14. Indeed, only the *Advanced administrator* will still be able to access all data contents (list in Table 7.13) from both processes *P2* and *P3*. For the *Administrator* and *Operator*, data contents $dc4.1$ and $dc5.1$ are only accessible through *P2* to investigate alerts, whereas event monitoring tasks (in *P3*) only expose the "sub-contents" $dc4.2$ and $dc5.2$, which, we will later see, has a lower risk level than $dc4.1$ and $dc5.1$ when exposed to these adversaries.

Table 7.14: Data content identification table (Privacy-aware TDS)

Exposure				**Data content**
Exp.	**Component**	**Adversary**	L_a	
$Exp1$	P2. Alert Investigation	Advanced administrator	Authorized	$dc1$, $dc2$, $dc3$, $dc4.1$, and $dc5.1$
$Exp2$	P2.	Administrator	Authorized	*Same data contents as above*
$Exp3$	P2.	Operator	Authorized	*Same data contents as above*
$Exp4$	P3. Event Monitoring	Advanced administrator	Authorized	$dc1$, $dc2$, $dc3$, $dc4.2$, and $dc5.2$
$Exp5$	P3.	Administrator	Authorized	*Same data contents as above*
$Exp6$	P3.	Operator	Authorized	*Same data contents as above*

7.3.4 EPIC Step 4 (Privacy-aware TDS)

Threats $Th1$, $Th2$, and $Th3$ (threats identified at the level of process *P2*) risk levels were not affected by the adoption of the privacy-aware risk-based access control solution (observable by comparing Tables 7.10 and 7.17). This can be explained by the fact that adversary (or adversaries with this role *Advanced administrator*) usually has a very high trust level and the fact that the context requires a more permissive access to data to react to a potential security emergency. Therefore our access control system will not take restrictive decisions regarding requests issued by this adversary in this context. Consequently the re-identifiability (observable through L_{rid} see Tables 7.8 and 7.15), the violation magnitude, and the sensitivity (see Tables 7.9 and 7.16) of the data remain the same for these threats, whether we use our privacy-aware risk-based access control solution or not.

For similar threats $Th16$, $Th17$, and $Th18$ (threats exposing data contents $dc1$, $dc2$, and $dc3$ to the same adversary *Advanced administrator* in a different context "event monitoring") the overall risk level slightly dropped for the privacy-aware TDS using our access control approach (see Tables 7.10 and 7.17). Although the adversary (*Advanced administrator*) is very trusted, some of the data (the most sensitive) will only be released

Table 7.15: Privacy violation likelihood table (Privacy-aware TDS)

Exposure	**Data content**	L_a	L_{rid}	Th	L
$Exp1$: P2 Advanced admin.	$dc3$	Authorized	Medium	$Th1$	**High**
	$dc2$	Authorized	High	$Th2$	**Very High**
	$dc3$	Authorized	High	$Th3$	**Very High**
$Exp2$: P2 Administrator	$dc4.1$	Authorized	Medium	$Th9$	**High**
	$dc5.1$	Authorized	High	$Th10$	**Very High**
$Exp3$: P2 Operator	$dc4.1$	Authorized	Medium	$Th14$	**High**
	$dc5.1$	Authorized	High	$Th15$	**Very High**
$Exp1$: P3 Advanced admin.	$dc3$	Authorized	Medium	$Th16$	**Medium**
	$dc2$	Authorized	High	$Th17$	**High**
	$dc3$	Authorized	High	$Th18$	**High**
$Exp2$: P3 Administrator	$dc4.2$	Authorized	High	$Th24$	**Medium**
	$dc5.2$	Authorized	Certain	$Th25$	**High**
$Exp3$: P3 Operator	$dc4.2$	Authorized	High	$Th29$	**Negligible**
	$dc5.2$	Authorized	Certain	$Th30$	**Medium**

Table 7.16: Qualitative privacy violation impact table (Privacy-aware TDS)

Exposure	**Data content**	**Th**	**Violation magn.**	I_C	I_R	I
$Exp1$: P2 Advanced admin.	$dc1$	$Th1$	Limited	Low	Low	**Low**
	$dc2$	$Th2$	Very-limited	Low	Low	**Low**
	$dc3$	$Th3$	Important	Low	Medium	**Medium**
$Exp2$: P2 Admin.	$dc4.1$	$Th9$	Very-limited	Medium	Medium	**Med-high**
	$dc5.1$	$Th10$	Very-limited	Medium	Medium	**Med-high**
$Exp3$: P2 Operator	$dc4.1$	$Th14$	Very-limited	Med-High	Med-High	**High**
	$dc5.1$	$Th15$	Very-limited	Med-High	Med-High	**High**
$Exp4$: P3 Advanced admin.	$dc1$	$Th16$	Medium	Low	Low	**Low**
	$dc2$	$Th17$	Medium	Low	Med-low	**Med-low**
	$dc3$	$Th18$	Important	Low	Medium	**Medium**
$Exp5$: P3 Admin.	$dc4.2$	$Th24$	Medium	Med-low	Med-low	**High**
	$dc5.2$	$Th25$	Limited	Med-low	Med-low	**High**
$Exp6$: P3 Operator	$dc4.2$	$Th29$	Medium	Medium	Med-low	**High**
	$dc5.2$	$Th30$	Limited	Medium	Med-low	**High**

with some adjustments (e.g., anonymization to lower the risk, monitored access to enhance trust).

Threats $Th9$, $Th10$, $Th14$ and $Th15$ expose data contents $dc4.1$ (in $Th9$ and $Th14$) and $dc5.1$ ($Th10$ and $Th15$) by the process *P2* (in the context of alert investigation) to two different actors (*Administrator* and *Operator*). The likelihood of re-identification L_{rid} dropped significantly for both involved actors (*Administrator* and *Operator*) from {*high*,

certain} to {*medium*, *high*} which makes the overall likelihood of violation L drop (in an observable way for $Th9$ and $Th14$, see difference between Table 7.8 and Table 7.15). This decrease is due to the fact that our privacy-aware risk-based access control system is expected, even in this context, to enforce anonymization (more or less strong anonymization depending on the trust of the actor/requester) on $dc4.1$ and $dc5.1$, before releasing the information, because these data contents are very easily re-identifiable (they contain attributes like *Email(header)*) and very sensitive (contain sensitive information e.g., *Email(content)* and *File(content)*). The magnitude of violation decreases as well and it becomes *very-limited* for all four threats, in fact, as mentioned earlier, the adoption of our access control system divided $dc4$ and $dc5$ (see Table 7.5) to "sub-contents" $dc4.1$, $dc5.1$ only exposed in case of alerts and $dc4.2$ and $dc5.2$ exposed as events. $dc4.2$ and $dc5.2$ contain less sensitive attributes (see Table 7.13), but have higher number of records with leave $dc4.1$ and $dc5.1$ with a limited number of records. This also reflected on the impact severity of the threats, which dropped from (see Tables 7.9 and 7.16) *med-high* to *medium* for $Th9$ and $Th14$ (threats involving the actor *Administrator*) and from *high* to *med-high* $Th10$ and $Th15$ (threats involving the actor *Operator*) the operator accessing this very sensitive data (even in a security context) is still problematic and have high impacts on both compliance and reputation the data accessed was used to violate privacy. The overall risk for these threats

$Th24$, $Th25$, $Th29$, and $Th30$ are threats identified at the level of *P3*. They now expose data contents $dc4.2$ and $dc5.2$ (instead of $dc4$ and $dc5$ see Tables 7.7 and 7.14) to the actors *Administrator* and *Operator* in the context of event monitoring. As mentioned earlier $dc4.2$ and $dc5.2$ are sub-sets of $dc4$ and $dc5$ and containing less QIDs and sensitive attributes (e.g., no *File(content)* in $dc4.2$ and no *Email(content)* in $dc5$). These attributes are not supposed to be accessed in the context of event monitoring, and if really needed the *Administrator* and *Operator* have to fulfill trust enhancement actions to be granted access. These changes in the data contents in addition to anonymization applied to high risk views of $dc4.2$ and $dc5.2$, drastically decrease the likelihood of re-identification L_{rid} and the overall privacy violation likelihood L of these threats (see Tables 7.8 and 7.15). The magnitude of the violation does not observably decrease (since the quantity of data is still very important and contains some sensitive information e.g., Email(header), File(meta) that can still be used to violate privacy). Nonetheless, impact factors I_C and I_R and the overall I decrease significantly. Consequently, the adoption of our privacy-aware risk-based access control system decreased significantly the over all risk for threats $Th24$, $Th25$, $Th29$, and $Th30$ (see Tables 7.10 and 7.17)

Table 7.17: Qualitative privacy violation risk table (Privacy-aware TDS)

Th	**Exposure**	**Data content**	L	$\mathcal{I}$	R
$Th1$	$Exp1$: P2 Advanced admin.	$dc1$	High	Low	**Low**
$Th2$	$Exp1$: P2 Advanced admin.	$dc2$	Very-high	Low	**Medium**
$Th3$	$Exp1$: P2 Advanced admin.	$dc3$	Very-high	Medium	**High**
$Th9$	$Exp2$: P2, Administrator	$dc4$	High	Medium	**High**
$Th10$	$Exp2$: P2, Administrator	$dc5$	Very-high	Medium	**Medium**
$Th14$	$Exp3$: P2, Operator	$dc4$	High	Med-high	**High**
$Th15$	$Exp3$: P2, Operator	$dc5$	Very-high	Med-high	**High**
$Th16$	$Exp4$: P3 Advanced admin.	$dc1$	Medium	Low	**Low**
$Th17$	$Exp4$: P3 Advanced admin.	$dc2$	High	Med-Low	**Medium**
$Th18$	$Exp4$: P3 Advanced admin.	$dc3$	High	Medium	**Medium**
$Th24$	$Exp5$: P3, administrator	$dc4.2$	Medium	Med-Low	**Low**
$Th25$	$Exp5$: P3, administrator	$dc5.2$	High	Med-Low	**Medium**
$Th29$	$Exp6$: P3, Operator	$dc4.2$	Negligible	Medium	**Low**
$Th30$	$Exp6$: P3, Operator	$dc5.2$	Medium	Medium	**Medium**

7.4 Chapter conclusions

In this chapter, we evaluated the privacy-aware access control approach from the privacy perspective by using the EPIC methodology (described in Chapter 3). To this scope, we applied the methodology to identify and evaluate privacy threats originated by authorized insider actors for two cybersecurity systems: a) a "classic TDS" and a the same TDS equipped with our privacy-aware access control "privacy-aware TDS". Then, we compared the risk values of threats identified in both systems.

The overall mitigatory impact of our approach (privacy-aware access control approach) on authorized insider threats' privacy risk level can be seen by comparing Table 7.10 and Table 7.17. Integrating our approach within a privacy-aware TDS results in a significant decrease of the risk levels of threats where sensitive data was accessed in contexts where there no "strong" need to access it. Thus, it reinforces the application of the data minimization principle and grants access on real need-to-know bases. The adoption of our approach did not have an observable impact on threats when data are accessed for analyzing a security alert. Indeed in this context, the security needs outweigh the sensitivity of the data and legitimize access especially for trusted actors. At a more granular level, our privacy-aware access control approach decreases the magnitude of exposures even in this context, by 1) decreasing the magnitude of violation by decreasing access to sensitive data contents 2) decreasing the likelihood of re-identification by limiting access to identifying attributes (e.g., through anonymization) and consequently the total privacy-violation likelihood. 3) mitigating the impact of privacy violations from both legal (non-compliance impact factor) and reputational (reputation loss impact factor) aspects. these impacts

are more observable for categories of actors with low trust level (i.e., Operators).

One of the aspects that we did not consider in this evaluation the impact of our access control models on privacy threats involving access to data in a non-trusted context (e.g., from a personal mobile). This is mainly because this scenario is very unlikely in the context of corporate cybersecurity. It is indeed rare that a company allows security agents to carry on critical security tasks, such as attack investigation, and forensic from a personal laptop or even a corporate mobile. However, if this scenario was to be, our methodology (EPIC) would model these two situations (i.e., access from a trusted device and access from a non-trusted device) as two different data flows (in the DFD+ at Step1 see 3.2.2) through different components with different security mechanisms (see Table 3.1) and involving two different actors (see Table 3.2) with different trust level. This way we can also capture the impact of using our privacy-aware risk-based access control in these scenarios.

Chapter 8

Industrial Impact

*This thesis is part of a the European Industrial Doctorate on **SEC**urity and Trust of Next Generation **ENT**erprise **I**nformation **S**ystems (**SECENTIS**) held in collaboration with the industrial partner SAP. Therefore this work is partly motivated by industrial applications. In this Chapter, we discuss the impact of our work in terms of possible migration to industry, standardization bodies, and open source communities.*

8.1 Introduction

Businesses are increasingly leveraging data to provide their customers with more efficient, more customized and faster services and products. (as epitomized by the famed Economist's article: *Worlds most valuable resource is no longer oil, but data*). In fact, by collecting and analyzing more data, companies are now in the position to improve their products, which, in turn, can attract more users, and generating even more data, and so on.

The access to massive and granular data can provide a company a competitive advantage, but it also imposes significant burden on the management and governance of this (often) confidential data. This is particular true, when we deal with personal data, which are highly regulated, and the desirable utility of data access should be carefully balance with the privacy risk. As a consequence, companies need processes and technologies to carefully assess and control privacy risk, and, if needed apply risk mitigation measures to limit the risk exposure.

As the market leader in enterprise software applications, SAP is also working to address these concerns. Indeed, SAP is focusing on developing novel applications, which fully exploit the competitive advantage (in terms of performance) of cutting-edge technologies such as HANA its in-memory database. At the same time, SAP supports its customers to comply with data protection regulations and the increasing privacy awareness of their

users, especially in the light of the new European data protection regulation (GDPR).

Therefore both research axes covered by the thesis (privacy threat assessment and risk-aware access control as a privacy-enhancing solution) are relevant for the software industry as well as for SAP. More in details, threat modeling is considered one the key element of SAP secure development lifecycle, and although originally focusing on security threats, it is including more and more data protection aspects. Indeed privacy threat identification and evaluation is one of the requirements of the new GDPR regulation, for all organizations dealing with personal data (which is almost the case of all modern organizations). The methodology described in Chapters (REF) provides a valuable instrument for privacy threat modeling and assessment of risk, which fits the SAP risk-based approach for secure software development.

The balance between data access and privacy is also of primary importance for SAP and SAP customers. classic access control systems (currently used) can cope with most of the existing business scenarios, but they can do that at the price of using complex and ad-hoc security policies, and in perspective more flexible and intuitive access controls systems are desirable. The business decision process is driven by risk assessment, accordingly, it appears natural considering access control system based on risk evaluation. In this context, the examples presented here represent a significant playground for testing novel access control methods on SAP technologies. Indeed using our privacy-aware risk-based access control allows for optimizing the trade-off between privacy and data exploitation. We also showed how a calibrated application of anonymization can be used to reduce the risk. SAP has a relatively long history in research on anonymization [154], and, more recently, it has been releasing anonymization capabilities in its products [137].

In this chapter, we will emphasize the relevance of this thesis work in the industrial environment and particularly for our industrial partner SAP. In Section 8.2 we will describe more in detail to some usage scenarios concerning both "EPIC" and the privacy-aware risk-based access control. In Section 8.3 we will discuss existing standardized relevant to this thesis and in Section 8.4 we will present some open source tools used in our research.

8.2 Industrial Use cases

8.2.1 Processes and automation for privacy impact assessment

Preserving users privacy has become a major concern for every organizations and businesses in the last decade. Indeed an IBM report [96] estimates a $4 million average cost of a privacy breach in 2016. In 2018, the European Unions General Data Protection Regulation (GDPR) will introduce fines of to 20 million euros or 4% of annual worldwide turnover for companies failing to meet the GDPR requirements [57]. A recent research [97] reveals that an organization expect to spend over one and a quarter million euros (€1,360,567 or

$1,432,176) in order to achieve full compliance. This amount is obviously well below the possible cost of non-compliance, however, it is also a strong call for methods and tools to support privacy threat modeling and risk assessment for GDPR compliance.

Risk management is of primary importance at SAP. At global level, SAP has established comprehensive internal control and risk management structures that enable the company to identify and analyze risks early, and take appropriate actions [139] This system has numerous control mechanisms and it is an important element of SAP corporate decision-making process; it is therefore implemented as an integral part of SAPs business processes across the entire SAP group and SAP's business. The risk-based approach is also at the core of SAP security. Starting from 2012, SAP established a threat modeling process and it is now a standard component in SAPs Secure Software Development Lifecycle. Originally based on the STRIDE threats list [124], it is increasingly integrating new threats, and especially privacy threats.

The increasing complexity and specialization of software systems (and corresponding privacy threats) calls for more sector-specific threat assessment. In this context, the EPIC methodology, described in Chapter 3, provides a strong guidance for the adaptation and extension of privacy threat models specific for cybersecurity systems. SAP is largely investing for keeping its track record of safeguarding businesses against security and privacy threats, continuously innovating the security features of its product portfolio, but also releasing specific security products such as SAP Enterprise Threat Detection (SAP ETD), SAP Governance, Risk and Compliance solution (GRC), SAP Identity Management. The secure development of all cybersecurity products needs a careful and specific threat modeling, and the large presence of personal data, such as log-entries (SAP ETD) or identity information (SAP Identity Management), give a predominant role to specific privacy threats.

Lastly, we have to mention that structured approach of EPIC methodology makes it particularly suitable for being implemented as a tool to increase the automation of the whole process, along the lines of the customization of SAP GRC for privacy impact assessment, which was recently pioneered by SAP Security Research [3, 49].

8.2.2 Privacy preserving threat detection

As described in the use-cases, presented in Chapter 5, Section 5.5, the risk-based access control model could be applied to enhance data access for threat detection systems.

SAP has a solution to analyze security log files for detecting possible intrusions, SAP Enterprise Threat Detection (SAP ETD) [140]. This product was originally prototyped by SAP Security Research, which has still a strict interaction with the product team. Accordingly, it is not surprising that our research has been conducted in close interaction with the product team.

SAP ETD collects application level security-relevant log data from SAP and non-SAP systems and analyzes the collected information in real-time in order to automatically detect attack patterns and generate alerts.

The logs often contain personal information (e.g., user-ids, IPs) together with information on the behavior of users (which, in case of employees, is strictly regulated by labor laws). The current version of ETD uses a pseudo-anonymization of User IDs as basic privacy enhancement technique to fulfill the data minimization requirement. This method provides privacy protection, but in complex scenarios, it may suffer from the existence of several identifiers and other elements that could be combined to re-identify or infer information about users (e.g., MAC and IP addresses, Terminal IDs, Timestamps).

In our research, we proposed a model permitting to devise a privacy-preserving access to logs (see Figure 8.1), keeping risk under control, and, at the same time, limiting the impact on the utility, as we showed in [110] testing our approach with real data from ETD infrastructure. In the same context, we also proposed tailored anonymization techniques applicable to security log-files.

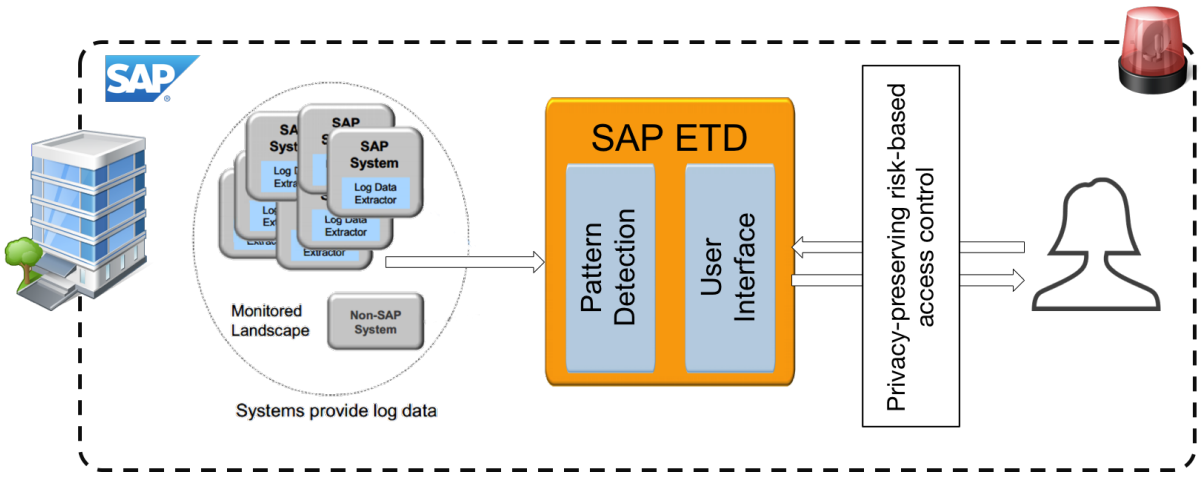


Figure 8.1: Single domain log files sharing

The results were presented to SAP DKOM event, which is the main SAP development community event, gathering representatives from a broad range of SAP development teams.

Furthermore, the ETD system is in premises solution: a company running an SAP ecosystem and the threat detection system, collects and use internal anonymized data to fulfill the privacy requirement described above, but the data stay *within the same company*, or in other words, they remain in the same trusted domain (see Figure 8.1). On the other hand, the access to logs from multiple domains (say, different companies/organizations) can help to detect more complex attacks, but it also increases tremendously the requirements from data protection and privacy point of view. Our research can support

the implementation of multi-trust domain logs access (as depicted in Figure 8.2), it could be realized in our model by setting different trust levels (i.e., different risk threshold) based in the trust relationship between the parties, for instance, a long-term partner organization may be more willing to share data (so high trust threshold), whereas other parties may have stronger privacy requirements and choose a more conservative approach. The usage of obligations, as proposed in Chapter 4 can increase the flexibility of the solution, allowing for handling more complex scenarios, such as considering geographical locations (geographies are extremely relevant for applicability of privacy laws).

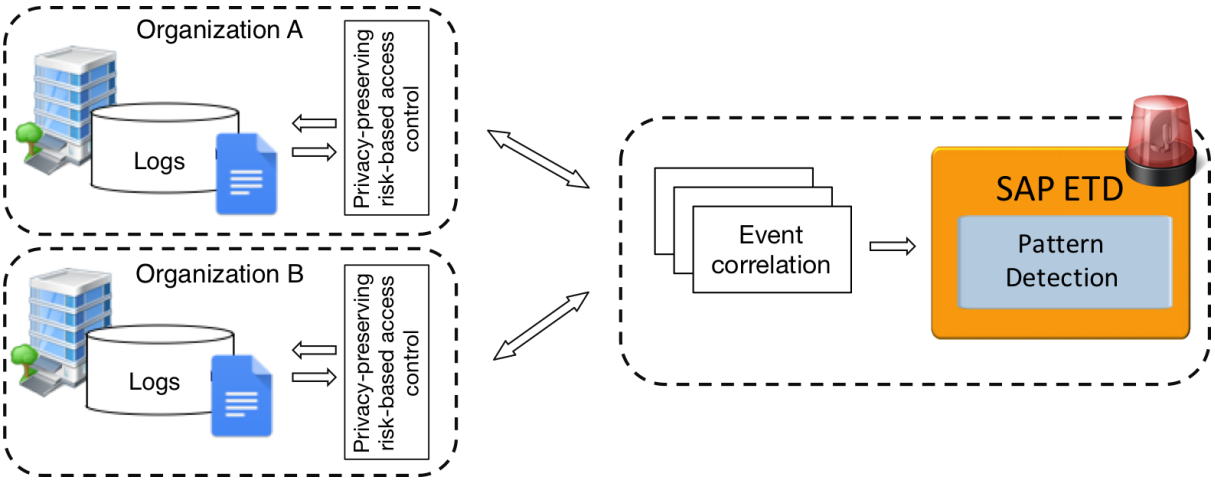


Figure 8.2: Miltiple-trust domain log files sharing

Although Threat Detection Systems are the prominent application example in the thesis, we should note that the model could be applied to other solutions, for example, to address complex authorization scenarios for Human Resource solutions (see the example use-case described in Chapter 6) or more general reporting use-cases, including mobile solutions (see the example use-case described in Chapter 4).

Lastly, a key element of our research is the usage of anonymization for reducing the risk and increasing the access of the data. Due to its relevance for multiple business cases, anonymization has considerably been investigated by SAP Security Research, both in the context of k-anonymity family models [154] and differential privacy models, and SAP offers multiple solutions with anonymization features: SAP TDMS [138], Anonymization Service for SAP Cloud Platform [137], and the above-mentioned SAP ETD. Our solution can leverage the features of SAP technologies, indeed, in many cases, querying a large database and extracting an anonymized dataset in real-time is very hard, and most anonymization processes are run off-line (i.e., as a batch process). However, exploiting the efficiency of in-memory databases, combined with column-store optimized algorithms as provided by SAP Hana would facilitate the implementation of an on-the-fly, flexible Privacy Risk Aware Access Control model that can be easily integrated with new data-intensive business

applications.

8.3 Standardization Bodies

There are no standards yet for privacy-aware and risk-Based access control systems. However, during the conception of our Risk-Based Access Control framework, as well as the implementation of prototypes we took reference from several well-known standards such as: the eXtensible Access Control Markup Language (XACML). XACML is an OASIS standard for fine-grained authorization management. XACML defines both an attribute-based policy language, and the architecture and dataflow model describing how to evaluate access requests and enforce access decisions according to the rules defined in policies. 5 In particular, our architecture (see Figure 4.2), is reminiscent of XACML proposal, and part of our results may be considered as an extension of the standards.

In addition, there are standards for security risk, such as the ISO/IEC 27000 family, and an access control based on risk estimation could be beneficial for implementing security control and mitigation measures for those standards.

8.4 Open-source Software

To the best of our knowledge no open source implementation, of any kind risk aware access control system, has been proposed yet (neither as a proprietary solution). However, several open source implementations of different models of access control system have been proposed. The most interesting propositions for as were "balana API" by Wso2 [162], HerasAF [53], and "ALFA" [11] a free closed source API by axiomatics.

For the implementation of the privacy-aware and risk-based Access control system we used open source Privacy Enhancement libraries available such as the java anonymization toolkit ARX [128]

Chapter 9

Conclusions and Future work

9.1 Conclusions

In this dissertation, we motivated, designed, implemented, and validated a novel privacy-aware risk-based access control model.

We develop and validate EPIC (Evaluating Privacy violation rIsk in Cyber-security systems), a privacy threat identification and evaluation methodology for cybersecurity systems. Since it is very difficult to follow a unique methodology to carry a privacy threat assessment in a general context [161] we chose to focus assessing privacy threats in cyber cybersecurity systems. This choice was motivated by the two facts a) these systems are a vital component of any information system; b) current cybersecurity products collect huge quantities of sensitive information and become increasingly invasive which makes a privacy threat assessment on these systems a very interesting and useful exercise. We use EPIC to study different privacy threat scenarios in the cybersecurity systems of an organization and emphasize the importance of dealing with insider privacy violation threats since the threats have high risk and priority levels.

We propose, implement, and evaluate a novel access control model that integrating trust with risk and supports a flexible access control in dynamic contexts. trust and risk adjustment strategies are applied prior to access, in parallel with the resource consumption to ensure an acceptable level of risk.

We adapt this trust and risk-based access control model to the context of privacy and propose a privacy-aware risk-based access control.In this model, we provide a concrete way to quantitatively estimation of privacy risk based on a category of metrics called "syntactic anonymity metrics". To evaluate the feasibility and effectiveness of this approach we selected two case studies namely "*HR information disclosure*" and "*Privacy aware threat investigation*". We developed two prototypes based on a slightly different version of the framework and run a set of experiments on each implementation. The obtained results show that the framework leads to meaningful results and real-time performance for both

case studies.

We propose a second version of the privacy-aware risk-based access control using another category of privacy metrics, equally interesting "the differential privacy" metrics. This differential privacy based model allows for data access at different privacy levels, generating a sanitized data set according to the privacy clearance (trustworthiness) of a request. A first experimental analysis, considering an HR related use case, and a benchmarking dataset, indicates that the model can address complex privacy and utility requirements. Indeed, in our use case, we use this model to prevent classification model from violating the privacy of individuals in the training data while ensuring a decent level of accuracy to allow different actors to exploit the results of this analyses. The performance of this model, however, did not meet the real-time requirements and several improvements are still to be added to reach this goal.

We evaluate the privacy-aware access control approach impact on privacy by using the EPIC methodology. We do so by comparing the privacy violation risk of different threats identified in threat detection system (TDS), before and after integrating our privacy-aware access control to the TDS. The evaluation results show the mitigatory impact of our model on the threats originated by insider authorized actors.

Since this work is part of SECENTIS the European Industrial Doctorate on Security and Trust of Next Generation Enterprise Information Systems the **Chapter 8** of this thesis was dedicated to discussing the industrial impact of the Ph.D. thesis.

9.2 Future work

Some aspects of the work presented in the thesis could be further developed as future work. For instance in Chapter 6, we present a "differential privacy based" privacy-aware risk-based access control model; differentially private mechanisms suffer from privacy protection degradation [80] (i.e., privacy guarantees are lower) when the dataset is queried multiple times (by the same user, or several users if we consider a collusion scenario [43]). It would be interesting to develop metrics assessing this degradation in our model but also explore how our model can mitigate (if not reverse) this degradation when evaluating access decision for multiple queries.

The model we proposed (in Chapter 6) does not support yet this kind of scenarios and doesn't consider queries history. Although this might be sufficient for some cases (similar to the case study used to assess this model Section 6.2), in the majority of cases users are allowed to query the dataset multiple times. Therefore, more investigations and modifications are needed to make this model compatible with such scenarios.

Moreover, several possible future research directions can also be explored based on the work done in this thesis.We will describe three of them in this section: *i*) Improve the

trust and risk assessment (in Section 9.3).*ii*) a cryptographically enforced privacy-aware risk-based access control a distributed access control handling new software architectures and usage scenarios (in Section 9.4). *iii*) Develop privacy metrics and privacy management tools enabling the data-owner to be more involved in the access control process(in Section 9.5).

9.3 Improve the Trust and Risk Assessment

In Chapter 5 and Chapter 6, we show how it is possible, using some heuristics, to derive sound relative estimation (i.e., using dimensionless units) for trust and risk, in some specific usage scenarios. However a general approach applicable to multiple use cases is still missing.

Ideally, we should estimate trust and risk in terms of monetary value, which has several advantages: 1) it provides a common *unit of measure* to combine risk and trust factors of very different nature (e.g., security risk, compliance risk, privacy risk or trust from reputation systems, trust-factors, behavioral analysis), 2) it is easy to understand for non-technical experts 3) it can be easily combined with risk mitigation and trust enhancement strategies that have a clear monetary value (e.g., insurance, certifications, legal contracts, trusted devices).

Regarding the ***Risk*** assessment, in the short term, we would like to validate our risk-based access control model on other use cases, where some, quantitative methods are, even partially, available (ideally using monetary values). In this respect, it is particularly interesting to investigate emerging cyber-insurance models (building on techniques derived by the financial sector, e.g. Value-at-risk, Monte-Carlo simulations) to compute the values of cyber-risk and hence the cost of insurance premiums [143].

In terms of ***Trust*** assessment, as hinted in Chapter 5 we would also like to investigate the impact of authentication mechanisms on trust. This assessment is particularly relevant in cloud ecosystems such as the example presented in 9.4. Based on this estimated (i.e., probability of authentication success [83]) we should be able to significantly improve the trust assessment and implement optimal trust enhancement strategies. These strategies could include a combination of multiple authentication methods according to the risk associated with the request.

9.4 Cryptographically Enforced privacy-aware risk-based access control

The risk-based access control model we present in Chapter 4 (as well as the privacy-aware access risk-based control model in Chapters 5 and 6), is operating under the assumption that the party storing the data (and handling the risk-based access control process) is "fully" trusted. However, in more and more frequent cases this assumption does not hold anymore.

Let's consider, for instance, an e-Health service that monitors the health status of elderly people patients suffering from cognitive troubles. This monitoring is based on behavioral analyses using data collected in the patient's home. The service detects behavioral anomalies that can endanger the health of the patient (e.g., missing a prescribed medication for several days) and sends evaluation reports to the treating physician. It also allows family members to check on the patient. finally, the serves can detect emergency situations (e.g. potential fire hazard, the patient fell, wrong medication) and send alerts for quick interventions (e.g. to emergency services, firefighters).

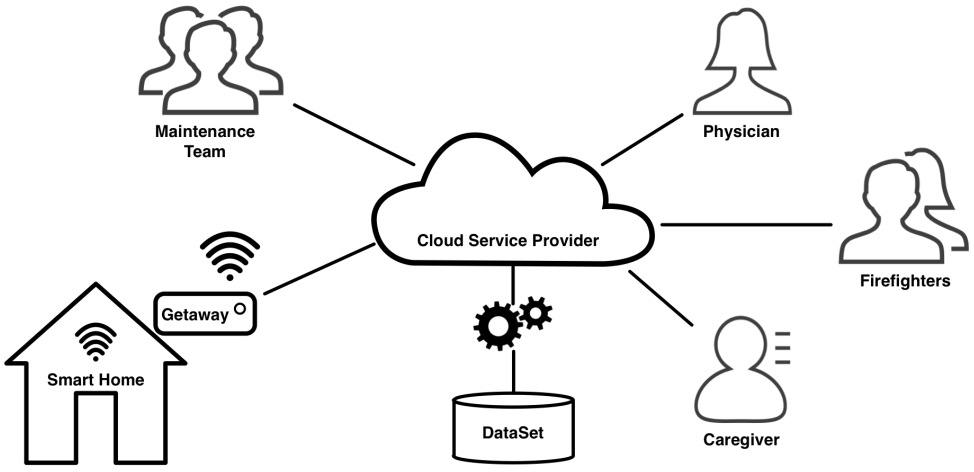


Figure 9.1: Smart-Home Behavioral Analysis Systems

Technically speaking, the service relies on a pervasive sensing infrastructure is deployed in each monitored home. The sensing devices unobtrusively capture the interaction of the inhabitant/patient with his/her surrounding environment. These raw measurements are then sent to a *gateway*, combined, analyzed and translated humanly readable information (e.g. temperature in a room, open/closed doors/repositories, manipulations of objects) called *events*. An *event* is represented by the type, the status and the time-stamps at which the event occurred. Due to the huge amount of data produced by the system, the gateway periodically (according to a user-defined time interval) transmits collected

data (i.e., events, ADLs, and anomalies) to a external storage hosted by a cloud service provider CSP. Data can then be queried and analyzed by authorized actors. Figure 9.1 illustrates a high-level architecture of the service as well as different actors interacting with (this model was inspired by a similar model proposed in [131]).

It is easy to see that this e-Health service is very privacy-invasive (i.e., permanent monitoring), and the kind of data it collects and processes is very sensitive (e.g., health, habits, and preferences etc.). It is, therefore, very important to preserve the confidentiality of this data and to carefully control access requests, respecting data minimization principle and data owners privacy preferences. For instance, the CSP, although semi-trusted, should not be able to see the data. this confidentiality requirement could be satisfied by storing the data encrypted. However, it is not very convenient to retrieve and decrypt all the data each time an actor requests access. Ideally, actors should be able to query the data while revealing a minimum amount of information to the CSP. Different actors should also have access to different amount (or granularity) of the data depending on their roles (e.g., family members, doctor, firefighters), data owner preferences(e.g., notify only the designated a next of kin, change doctors), and the context of access (e.g., the security level of the device used, geographical location).

Some of requirements identified in this example can be fulfilled using cryptographic access control models(see [64, 159]). Along these efforts, the framework will protect the confidentiality of the data from unauthorized accesses (including the CSP) by encrypting the data at the data owner level, before sending the encrypted version to be stored. The selected encryption schemes should allow executing a set of search queries as well as some operation such as risk estimation and mitigation operations. *Search-able data storage* Moreover, this framework will also provide a flexible access to authorized actors (e.g., doctors, emergency services) to fulfill their health-care tasks, while protecting the data owner's privacy *Attribute Based Encryption.* This second feature will be implemented through risk-based approach ensuring the enforcement of the minimization principle and maintaining the privacy risk under an acceptable threshold. Finally, risk mitigation actions such as data anonymization can be performed at the CSP level (en encrypted data) using *homomorphic encryption schemes.*

Search-able data storage As mentioned earlier this feature task is to ensure the confidentiality of the data (from non-authorized actors including the CSP hosting the data) and at the same time allow authorized users to run search queries over this encrypted data. Several encryption schemes have been proposed in the literature to enable queries over encrypted data (see [23] for review) Most of these works focus on a specific category of queries (e.g., word search queries, number comparison queries). However, as suggested

in [127] some of these cryptosystems can be combined and used to encrypt and store several version of the data in a way that covers the category of queries needed by the actors. After encryption, the encrypted data-set, with this *new* schema, is stored at a cloud storage. Each user/application (authorized to query the data) is equipped with a proxy that translates the data requests and answers to enable querying the encrypted data-set. This "new" data-set is stored in a cloud storage with a *new* schema (pseud anonymity table and column names).

Privacy-aware access control The privacy-aware risk-based access control process, proposed in this framework can be executed in three main operations:

- *Identity/attributes management.* The identity and attributes management consists of collecting, updating, and suppressing authorized actors attributes. These attributes can describe the actor him/her self (e.g., role, name), it can describe the device/network used to send the access request, it can also describe the context of the request (e.g., geographical location, time of the request). This operation aims to control the integrity and availability of information useful for the other two operations (i.e., privilege management and). In fact, a subset of the attributes will be used to generate decryption keys that will allow the actors to access the data, another subset will be used during the access evaluation to assess the access risk.

- *Privileges management.* As its name indicates, privilege management, is the ensemble of steps allowing to assign, update, and revoke privileges to/from an actor. A privilege in our context is the ability to decrypt data attributes and access the message in clear. Which means privileged actors will possess a decryption key, and managing privileges comes back to managing the actors key (i.e., encrypting data and generating decryption keys) Different actors will have different privileges, and consequently different keys. However, we would also like the encryption to be done on a unique version (i.e., not an encrypted version for each user) and we would like it done by a unique key (possessed by the data owner). Attribute-based encryption or ABE presented in [64, 77], offers a natural solution to this situation. In fact this encryption model allows *i)* to generate a master key held at the data owner and used to encrypt the data, *ii)* to generate several user keys (using the data owner's master key), and *iii)* to define an access structure/policy and a set of attributes describing which key can decrypt which data. It is possible to express authorizations through attribute-based object-centric policies using Cipher-text attribute-based encryption (CP-ABE [18, 70]). A policy $\pi_o \in Pi$ is used for the encryption for each object $o \in O$ (with Pi the set of all policies defined by a data owner and O the set of objects possessed by the data owner). Each policy π_o uses a set of attributes A_π to describe set of privileged users U^* can decrypt the data and in which set of contexts C^* the

privilege applies.

$$\pi_o \leftarrow \bigvee_{\substack{u \in U^* \\ c \in C^*}} \Big(\bigwedge_{a_u \in A'_u} a_u \quad \wedge \quad \bigwedge_{a_c \in A'_c} a_c \Big)$$

With $A'_u \subseteq A_u$ a subset of a user's attributes and $A'_c \subseteq A_c$ a subset of a context's attributes. Indeed each user $u \in U$ is grated a *user key* generated using set attributes A_u that describes c and a set of attributes A_c describing a context $c \in C$ where the key can be used. Each key describes the user and context according to the description function:

$$\mathcal{D}(key) = \bigwedge_{a_u \in A_u} a_u \quad \wedge \quad \bigwedge_{a_c \in A_c} a_c$$

The decryption privilege applies for a user u in the context c *iff* $\pi_o \equiv \mathcal{D}(u) \wedge \pi_o \equiv \mathcal{D}(c)$

- *Access evaluation and enforcement.* Similarly to the model presented in Chapter 4 4.1, in this step the access control system, decides to grant access or not to the encrypted resource the requester based on the privacy risk (sensitivity and likelihood of re-identification of the requested data-view) and the trust of the request.

Risk and Trust evaluation and Adjustment Since the data is stored encrypted at the CSP level. The CSP should be able to provide a way to estimate the privacy risk. If we take the case of k-anonymity based re-identification risk the estimation of k the cardinality of a data-view (as described in Section 4.3.3) is based on a comparison of QIDs and can be computed using searchable encryiption [127]. For more elaborate risk evaluation over encrypted data, some work is still to be done by exploring the possibilities of using homomorphic encryption algorithms to compute over encrypted data (see [38, 39]). Same for data anonymization (as risk mitigation strategy) we would like to develop anonymization algorithms that could be executed over homomorphically encrypted data so the CSP can execute the adequate anonymization corresponding to the risk and trust level of a request and return an anonymized version of the data when needed, still without discovering the content of the data.

9.5 Data-owner centric privacy management

The third direction is to develop privacy metrics and privacy management tools based on a "crowdsourced" perception of privacy and enabling data-owners to be more involved controlling access to their data.

The idea is to provide data owners with tools to monitor their privacy and be able to grant (or deny access) to their data in a privacy-preserving way when this is possible. It

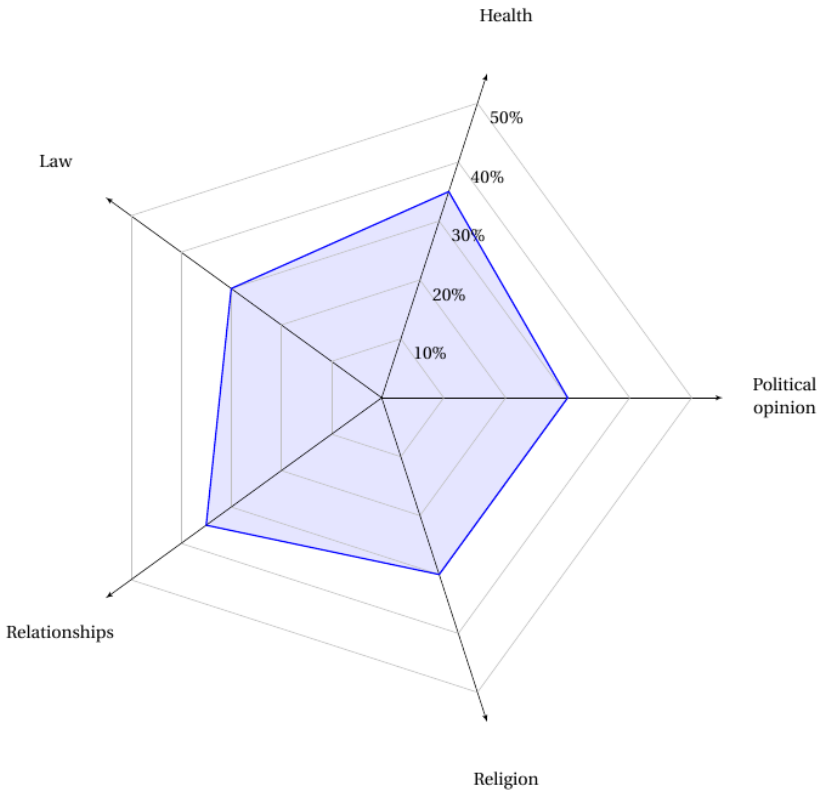


Figure 9.2: Privacy radar

also can be used to enforce *data-owners* privacy preferences when a third-party is handling access to their data. This can be achieved through the classification of the requested data into categories (e.g., Activism, Business, Health, Law, Politics, Relationships, Religion). Then we assess the privacy sensitivity of the data in each category (as shown in Figure 9.2).

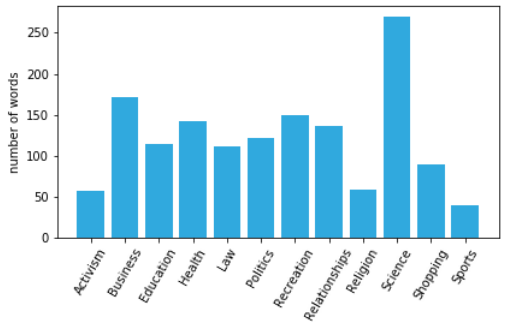


Figure 9.3: Privacy Lexicon: distributions of words by category

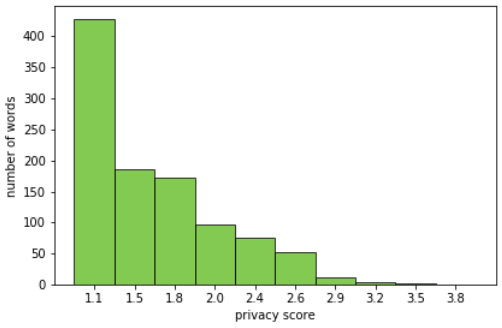


Figure 9.4: Privacy Lexicon: distributions of words by sensitivity level

These two operations will be carried using a crowdsourced privacy lexicon labeling a set of words with sensitivity and category tags (as inspired by sentiment analyses approach).

The access is then evaluated based on these sensitivity levels assessed against a sensitivity threshold set by the data-owner (or based on her/his preferences).

The first step toward this, is to build the privacy lexicon. We already started this operation using amazon mechanical turk [25] and we collected a first batch of 1030 words classified according to 12 categories and a sensitivity range between 0 to 4 (0 being the least sensitive and 4 the most sensitive). We are currently trying to enrich this privacy lexicon using lexicon extension techniques and will be soon releasing for public use. Figure 9.3 and 9.4 describe some characteristics of our lexicon in terms of categories and privacy sensitivity.

Bibliography

[1] A. Ahmed and N. Zhang. A context-risk-aware access control model for ubiquitous environments. In *IMCSIT*. IEEE, 2008.

[2] M. Ali, L. Bussard, and U. Pinsdorf. Obligation language for access control and privacy policies. , 2010.

[3] R. Alnemr, E. Cayirci, L. Dalla Corte, A. Garaga, R. Leenes, R. Mhungu, S. Pearson, C. Reed, A. Santana de Oliveira, D. Stefanatou, et al. A data protection impact assessment methodology for cloud. In *Annual Privacy Forum*, pages 60–92. Springer, 2015.

[4] C. Ardagna, S. "De Capitani di Vimercati", S. Paraboschi, E. Pedrini, P. Samarati, and M. Verdicchio. Expressive and deployable access control in open web service applications. *IEEE Transactions on Service Computing (TSC)*, 4(2):96–109, April-June 2011.

[5] C. A. Ardagna, M. Cremonini, S. De Capitani di Vimercati, and P. Samarati. A privacy-aware access control system. *Journal of Computer Security*, 16(4):369–397, 2008.

[6] C. A. Ardagna, E. Damiani, S. De Capitani di Vimercati, and P. Samarati. Towards privacy-enhanced authorization policies and languages. In S. Jajodia and Duminda Wijesekera, editors, *Data and Applications Security XIX*, pages 16–27, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

[7] C. A. Ardagna, S. De Capitani di Vimercati, G. Neven, S. Paraboschi, F. S. Preiss, P. Samarati, and M. Verdicchio. Enabling privacy-preserving credential-based access control with xacml and saml. In *2010 10th IEEE International Conference on Computer and Information Technology*, pages 1090–1095, June 2010.

[8] A. Armando, M. Bezzi, F. Di Cerbo, and N. Metoui. Balancing trust and risk in access control. In *On the Move to Meaningful Internet Systems: OTM 2015 Conferences*, pages 660–676. Springer International Publishing, 2015.

[9] A. Armando, M. Bezzi, N. Metoui, and A. Sabetta. Risk-aware information disclosure. In *Data Privacy Management, Autonomous Spontaneous Security and Security Assurance*, pages 266–276. Springer, 2015.

[10] A. Armando, M. Bezzi, N. Metoui, and A. Sabetta. Risk-based privacy-aware information disclosure. *Int. J. Secur. Softw. Eng.*, 6(2):70–89, April 2015.

[11] Axiomatics. Axiomatics language for authorization (alfa). `http://www.axiomatics.com/axiomatics-alfa-plugin-for-eclipse.html`. Accessed: 2017-10-15.

[12] N. Baracaldo and J. Joshi. A trust-and-risk aware RBAC framework: Tackling insider threat. In *Proceedings of the 17th ACM Symposium on Access Control Models and Technologies*, SACMAT '12, pages 167–176, New York, NY, USA, 2012. ACM.

[13] N. Baracaldo and J. Joshi. An adaptive risk management and access control framework to mitigate insider threats. *Computers and Security*, 39, Part B(0):237 – 254, 2013.

[14] N. Baracaldo and J. Joshi. An adaptive risk management and access control framework to mitigate insider threats. *Computers and Security*, 39, Part B:237 – 254, 2013.

[15] UC Berkeley. Privacy patterns. https://www.privacypatterns.org/. Accessed: 2017-07-14.

[16] S. Berthold and R. Böhme. Valuating privacy with option pricing theory. *Economics of information security and privacy*, pages 187–209, 2010.

[17] E. Bertino, P. A. Bonatti, and E. Ferrari. Trbac: A temporal role-based access control model. *ACM Trans. Inf. Syst. Secur.*, 4(3):191–233, 2001.

[18] J. Bethencourt, A. Sahai, and B. Waters. Ciphertext-policy attribute-based encryption. In *2007 IEEE Symposium on Security and Privacy (SP '07)*, pages 321–334, May 2007.

[19] C. Bettini, S. Mascetti, X. S. Wang, D. Freni, and S. Jajodia. Anonymity and historical-anonymity in location-based services. In *Privacy in Location-Based Applications: Research Issues and Emerging Trends*, pages 1–30. Springer, Berlin, Heidelberg, 2009.

[20] C. Bettini, X. S. Wang, and S. Jajodia. The role of quasi-identifiers in k-anonymity revisited. *CoRR*, abs/cs/0611035, 2006.

[21] M. Bezzi. An information theoretic approach for privacy metrics. *Transactions on Data Privacy*, 2010.

[22] P. Bonatti, C. Galdi, and D. E. Torres. Erbac: Event-driven rbac. In *Proceedings of the 18th ACM Symposium on Access Control Models and Technologies*, SACMAT '13, NY, USA, 2013. ACM.

[23] C. Bösch, P. Hartel, W. Jonker, and A. Peter. A survey of provably secure searchable encryption. *ACM Comput. Surv.*, 47(2):18:1–18:51, August 2014.

[24] J. Brickell and V. Shmatikov. The cost of privacy: Destruction of data-mining utility in anonymized data publishing. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, pages 70–78, New York, NY, USA, 2008. ACM.

[25] M. Buhrmester, T. Kwang, and S. D. Gosling. Amazon's mechanical turk: A new source of inexpensive, yet high-quality, data? *Perspectives on psychological science*, 6(1):3–5, 2011.

[26] M. Burdon and P. Harpur. Re-conceptualising privacy and discrimination in an age of talent analytics. *UNSWLJ*, 37:679, 2014.

[27] J. Byun, E. Bertino, and N. Li. Purpose based access control of complex data for privacy protection. In *Proceedings of the Tenth ACM Symposium on Access Control Models and Technologies*, SACMAT '05, pages 102–110, New York, NY, USA, 2005. ACM.

[28] J. Byun and N. Li. Purpose based access control for privacy protection in relational database systems. *The VLDB Journal*, 17(4):603–619, July 2008.

[29] N. Cao, Z. Yang, C. Wang, K. Ren, and W. Lou. Privacy-preserving query over encrypted graph-structured data in cloud computing. In *Distributed Computing Systems (ICDCS), 2011 31st International Conference on*, pages 393–402. IEEE, 2011.

[30] R. Caralli, J. Stevens, L. Young, and W. Wilson. Introducing octave allegro: Improving the information security risk assessment process. Technical report CMU/SEI-2007-TR-012, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, 2007.

[31] E. Celikel, M. Kantarcioglu, B. Thuraisingham, and E. Bertino. A risk management approach to RBAC. *Risk Decis. Anal.*, 1(1):21–33, 2009.

[32] CERT. "2014 u.s. state of cybercrime survey", 2014.

[33] CERT. "2017 u.s. state of cybercrime survey", 2017.

[34] L. Chen and J. Crampton. Risk-aware role-based access control. In Catherine Meadows and Carmen Fernandez-Gago, editors, *Security and Trust Management*, volume 7170 of *Lecture Notes in Computer Science*, pages 140–156. Springer Berlin Heidelberg, 2012.

[35] L. Chen, J. Crampton, M. J. Kollingbaum, and T. J. Norman. Obligations in risk-aware access control. In *PST*, pages 145–152, 2012.

[36] YL. Chen et al. Data flow diagram. In *Modeling and Analysis of Enterprise and Information Systems*, pages 85–97. Springer, 2009.

[37] PC. Cheng, P. Rohatgi, C. Keser, P. A. Karger, G. M. Wagner, and A. C. Reninger. Fuzzy multi-level security: An experiment on quantified risk-adaptive access control. In *IEEE Symposium on Security and Privacy*, pages 222–230. IEEE Computer Society, 2007.

[38] J. H. Cheon, M. Kim, and M. Kim. Search-and-compute on encrypted data. In *International Conference on Financial Cryptography and Data Security*, pages 142–159. Springer, 2015.

[39] J. H. Cheon, M. Kim, and K. Lauter. Homomorphic computation of edit distance. In *International Conference on Financial Cryptography and Data Security*, pages 194–212. Springer, 2015.

[40] V. Ciriani, S. De Capitani di Vimercati, S. Foresti, and P. Samarati. k-Anonymity. In T. Yu and S. Jajodia, editors, *Secure Data Management in Decentralized Systems*. Springer-Verlag, 2007.

[41] V. Ciriani, S. De Capitani di Vimercati, S. Foresti, and P. Samarati. Theory of privacy and anonymity. In M. Atallah and M. Blanton, editors, *Algorithms and Theory of Computation Handbook (2nd edition)*. CRC Press, 2009.

[42] R. Clarke. Privacy impact assessment: Its origins and development. *Computer Law & Security Review*, 25(2):123 – 135, 2009.

[43] C. Clifton and T. Tassa. On syntactic anonymity and differential privacy. In *2013 IEEE 29th International Conference on Data Engineering Workshops (ICDEW)*, pages 88–93, April 2013.

[44] CNIL (Commission Nationale de l'Informatique et des Libertés). *PRIVACY IMPACT ASSESSMENT (PIA): Methodology (how to carry out a PIA)*, June 2015.

[45] C. Colwill. Human factors in information security: The insider threat - who can you trust these days? *Information Security Technical Report*, 14(4):186–196, November 2009.

[46] CORDIS. Privacy and Data Protection Impact Assessment Framework for RFID Applications. Report, European Commission's Community Research and Development Information Service, 2011.

[47] T. Dalenius. Finding a needle in a haystack-or identifying anonymous census record. *Journal of official statistics*, 2(3), 1986.

[48] S. J. De and D. Le Métayer. Priam: A privacy risk analysis methodology. In *International Workshop on Data Privacy Management*, pages 221–229. Springer, 2016.

[49] A. Santana de Oliveira. Implementing data protection impact assessments with sap grc risk management. https://blogs.sap.com/2017/03/02/implementing-data-protection-impact-assessments-with-sap-grc-risk-management/. Accessed: 2018-01-15.

[50] M. Deng, K. Wuyts, R. Scandariato, B. Preneel, and W. Joosen. A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements. *Requirements Engineering*, 16(1):3–32, 2011.

[51] F. Di Cerbo, F. S. Doliere, L. Gomez, and S. Trabelsi. Ppl v2.0: Uniform data access and usage control on cloud and mobile. In *Proceedings of the 1st International Workshop on TEchnical and LEgal aspects of data pRIvacy and SEcurity*. IEEE, 2015.

[52] L. Dickens, A. Russo, PC. Cheng, and Jorge Lobo. Towards learning risk estimation functions for access control. In *In Snowbird Learning Workshop*, 2010.

[53] S. Dolski, Florian Huonder, and Stefan Oberholzer. Heras-af xacml core. `https://bitbucket.org/herasaf/herasaf-xacml-core`, 2016.

[54] C. Dwork. Differential privacy. In Michele Bugliesi, Bart Preneel, Vladimiro Sassone, and Ingo Wegener, editors, *Automata, Languages and Programming*, volume 4052 of *Lecture Notes in Computer Science*, pages 1–12. Springer Berlin / Heidelberg, 2006. 10.1007/11787006_1.

[55] C. Dwork. Differential privacy: A survey of results. In *International Conference on Theory and Applications of Models of Computation*, pages 1–19. Springer, 2008.

[56] ENISA. Privacy enhancing technologies: Evolution and state of the art. Report, European Union Agency For Network and Information Security, December 2016.

[57] European advisory body on data protection and privacy. Guidelines on data protection impact assessment (pia). Report, European Commission, Directorate General Justice and Consumers, April 2017.

[58] *eXtensible Access Control Markup Language (XACML) Version 3.0*, January 2013. `http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.pdf`.

[59] "European Union Agency for Fundamental Rights". Handbook on european data protection law. Technical report, Council of Europe European Court of Human Rights, 2014.

[60] M. Friedewald and R. J. Pohoryles. *Privacy and Security in the Digital Age: Privacy in the Age of Super-Technologies*. Routledge, 2016.

[61] D. Gambetta. Can we trust trust? In *Trust: Making and Breaking Cooperative Relations*, pages 213–237. Basil Blackwell, 1988.

[62] S. R. Ganta, S. P. Kasiviswanathan, and A. Smith. Composition attacks and auxiliary information in data privacy. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, pages 265–273, New York, NY, USA, 2008. ACM.

[63] G. Ghinita, P. Karras, P. Kalnis, and N. Mamoulis. Fast data anonymization with low information loss. In *Proceedings of the 33rd international conference on Very large data bases*, pages 758–769. VLDB Endowment, 2007.

[64] V. Goyal, O. Pandey, A. Sahai, and B. Waters. Attribute-based encryption for fine-grained access control of encrypted data. In *Proceedings of the 13th ACM Conference on Computer and Communications Security*, CCS '06, pages 89–98, New York, NY, USA, 2006. ACM.

[65] B. Hale. Estimating log generation for security information event and log management. white paper, Solarwinds, 2012.

[66] W. Hartzog and I. Rubinstein. The anonymization debate should be about risk, not perfection. *Commun. ACM*, 60(5):22–24, April 2017.

[67] D. Henshel, M.G. Cains, B. Hoffman, and T. Kelley. Trust as a human factor in holistic cyber security risk assessment. *Procedia Manufacturing*, 3:1117 – 1124, 2015. 6th International Conference on Applied Human Factors and Ergonomics AHFE 2015.

[68] S. H. Houmb, V. N. L. Franqueira, and E. A. Engum. Quantifying security risk level from cvss estimates of frequency and impact. *J. Syst. Softw.*, 83(9):1622–1634, September 2010.

[69] M. Howard and S. Lipner. *The Security Development Lifecycle*. Microsoft Press, 2006.

[70] P. Hu and H. Gao. Ciphertext-policy attribute-based encryption for general circuits from bilinear maps. *Wuhan University Journal of Natural Sciences*, 22(2):171–177, Apr 2017.

[71] International Data Corporation (IDC). The digital universe of opportunities rich data and the increasing value of the internet of things. , April 2014.

[72] International Data Corporation (IDC). European data market smart 2013/0063. , February 2017.

[73] Information Commissioner Office (ICO). Conducting privacy impact assessments code of practice. Report, Information Commissioner Office, 2014.

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

[75] ISO. Information technology - Security techniques - Guidelines for privacy impact assessment. Standard, International Organization for Standardization, Geneva, CH, June 2017.

[76] A. Josang, R. Ismail, and C. Boyd. A survey of trust and reputation systems for online service provision. *Decision Support Systems*, 43(2):618 – 644, 2007. Emerging Issues in Collaborative Commerce.

[77] A. Juels and M. Szydlo. Attribute-based encryption: using identity-based encryption for access control, 2004.

[78] Y. k. Lee, S. Lee, J. Hwang, B. Chung, and D. G. Lee. Anonymous access control framework based on group signature. In *2010 2nd International Conference on Information Technology Convergence and Services*, pages 1–5, Aug 2010.

[79] M. Kaempfer. scn.sap.com/community/security/blog/2015/03/04/sap-enterprise-threat-detection-and-siem-is-this-not-the-same, 2015.

[80] P. Kairouz, S. Oh, and P. Viswanath. The composition theorem for differential privacy. *IEEE Transactions on Information Theory*, 63(6):4037–4049, 2017.

[81] S. Kandala, R. S. Sandhu, and V. Bhamidipati. An attribute based framework for risk-adaptive access control models. In *ARES*, pages 236–241. IEEE, 2011.

[82] B. Karabacak and I. Sogukpinar. Isram: information security risk analysis method. *Computers & Security*, 24(2):147–159, 2005.

[83] P.G. Kelley, S. Komanduri, M.L. Mazurek, R. Shay, T. Vidas, L. Bauer, N. Christin, L.F. Cranor, and J. Lopez. Guess again (and again and again): Measuring password strength by simulating password-cracking algorithms. In *Security and Privacy (SP), 2012 IEEE Symposium on*, pages 523–537, May 2012.

[84] B. Kenig and T. Tassa. A practical approximation algorithm for optimal k-anonymity. *Data Mining and Knowledge Discovery*, 25(1):134–168, 2012.

[85] F. Kohlmayer, F. Prasser, C. Eckert, S. Kemper, and K. A. Kuhn. Flash: Efficient, stable and optimal k-anonymity. In *ASE/IEEE International Conference on Privacy, Security, Risk and Trust*, SOCIALCOM-PASSAT '12, DC, USA, 2012. IEEE Computer Society.

[86] F. Kohlmayer, F. Prasser, C. Eckert, and K. A. Kuhn. A flexible approach to distributed data anonymization. *Journal of Biomedical Informatics*, 50:62 – 76, 2014. Special Issue on Informatics Methods in Medical Privacy.

[87] A. Kounine and M. Bezzi. Assessing disclosure risk in anonymized datasets. In *Proceedings of the FloCon Workshop*, January 2009.

[88] K. Lakkaraju and A. Slagell. Evaluating the utility of anonymized network traces for intrusion detection. In *Proceedings of the 4th International Conference on Security and Privacy in Communication Netowrks*, SecureComm '08, pages 17:1–17:8, New York, NY, USA, 2008. ACM.

[89] J. Larkin. *Strategic reputation risk management*. Springer, 2002.

[90] J. Lee and C. Clifton. Differential identifiability. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, pages 1041–1049, New York, NY, USA, 2012. ACM.

[91] D. Leoni. Non-interactive differential privacy: a survey. In *Proceedings of the First International Workshop on Open Data*, pages 40–52. ACM, 2012.

[92] N. Li, T. Li, and S. Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. In *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*, pages 106–115. IEEE, 2007.

[93] N. Li, T. Li, and S. Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. In *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*, pages 106–115, April 2007.

[94] N. Li, W. Qardaji, and D. Su. On sampling, anonymization and differential privacy or, k-anonymization meets differential privacy. In *Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security*, ASIACCS '12, pages 32–33, New York, NY, USA, 2012. ACM.

[95] X. Li, F. Zhou, and X. Yang. A multi-dimensional trust evaluation model for large-scale p2p computing. *Journal of Parallel and Distributed Computing*, 71(6):837–847, 2011.

[96] Ponemon Institute LLC. 2016 cost of data breach study: Global analysis. Technical report, Benchmark research sponsored by IBM, June 2016.

[97] Veritas LLC. 2017 veritas gdpr report. Technical report, veritas Technologies Corporation, 2017.

[98] J. Luna, N. Suri, and I. Krontiris. Privacy-by-design based on quantitative threat modeling. In *Risk and Security of Internet and Systems (CRiSIS), 2012 7th International Conference on*, pages 1–8. IEEE, 2012.

[99] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1), March 2007.

[100] L. Martin. Getting smart about your workforce: Why analytics matter. Technical report, Oracle Corporation & CedarCrestone, September 2011.

[101] L. D. Martino, N. Qun, D. Lin, and E. Bertino. Multi-domain and privacy-aware role based access control in ehealth. In *2008 Second International Conference on Pervasive Computing Technologies for Healthcare*, pages 131–134, Jan 2008.

[102] S. Mascetti, N. Metoui, A. Lanzi, and C. Bettini. Epic: a methodology for evaluating privacy violation risk in cyber security systems. *Submitted to: Transactions on Data Privacy*, 2018.

[103] D. McClure and J. P. Reiter. Differential privacy and statistical disclosure risk measures: An investigation with binary synthetic data. *Trans. Data Privacy*, 5(3):535–552, December 2012.

[104] R. McGraw. Risk-adaptable access control (radac). *NIST Privilege (Access) Management Workshop*, 2009.

[105] D. H. Mcknight and N. L. Chervany. The meanings of trust. Technical report, , 1996.

[106] D. H. McKnight and N. L. Chervany. Trust and distrust definitions: One bite at a time. In *Trust in Cyber-societies: Integrating the Human and Artificial Perspectives*, pages 27–54. Springer, Berlin, Heidelberg, 2001.

[107] P. Mell, K. Scarfone, and S. Romanosky. Common vulnerability scoring system. *IEEE Security & Privacy*, 4(6), 2006.

[108] N. Metoui and M. Bezzi. Differential privacy based access control. In *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, pages 962–974. Springer, 2016.

[109] N. Metoui, M. Bezzi, and A. Armando. Trust and risk-based access control for privacy preserving threat detection systems. In *International Conference on Future Data and Security Engineering*, pages 285–304. Springer, 2016.

[110] N. Metoui, M. Bezzi, and A. Armando. *Risk-Based Privacy-Aware Access Control for Threat Detection Systems*, pages 1–30. Springer Berlin Heidelberg, Berlin, Heidelberg, 2017.

[111] Y. Miaoui, N. Boudriga, and E. Abaoub. Economics of privacy: A model for protecting against cyber data disclosure attacks. *Procedia Computer Science*, 72:569–579, 2015.

[112] K. Mivule and Blake A. A study of usability-aware network trace anonymization. In *Science and Information Conference (SAI), 2015*, pages 1293–1304. IEEE, 2015.

[113] N. Mohammed, R. Chen, B. C. M. Fung, and P. S. Yu. Differentially private data release for data mining. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, pages 493–501, New York, NY, USA, 2011. ACM.

[114] M. C. Mont and F. Beato. On parametric obligation policies: Enabling privacy-aware information lifecycle management in enterprises. In *Policies for Distributed Systems and Networks, 2007. POLICY'07. Eighth IEEE International Workshop on*, pages 51–55. IEEE, 2007.

[115] T. Moses et al. extensible access control markup language (xacml) version 2.0. *Oasis Standard*, 200502, 2005.

[116] A. Narayanan, J. Huey, and E. W. Felten. A precautionary approach to big data privacy. In *Data Protection on the Move*, pages 357–385. Springer, 2016.

[117] P. G. Neumann. *Combatting Insider Threats*, pages 17–44. Springer US, Boston, MA, 2010.

[118] NIST. Privacy risk management for federal information systems. Internal Report, National Institute of Standards and Technology , 2017.

[119] N. Notario, A. Crespo, Y. S. Martn, J. M. D. Alamo, D. L. Mtayer, T. Antignac, A. Kung, I. Kroener, and D. Wright. Pripare: Integrating privacy best practices into a privacy engineering methodology. In *2015 IEEE Security and Privacy Workshops*, pages 151–158, May 2015.

[120] M. C. Oetzel and S. Spiekermann. A systematic methodology for privacy impact assessments: a design science approach. *European Journal of Information Systems*, 23(2):126–150, 2014.

[121] M. C. Jason Prograrm Office. "horizontal integration: Broader access models for realizing information dominance, jsr-04-132". Technical report, , December 2004.

[122] Committee on Strategies for Responsible Sharing of Clinical Trial Data;. *Sharing Clinical Trial Data: Maximizing Benefits, Minimizing Risk.* National Academies Press (US), Washington (DC), 2015.

[123] A. Oprea, Z. Li, T. Yen, S. H. Chin, and S. Alrwais. Detection of early-stage enterprise infection by mining large-scale log data. In *Dependable Systems and Networks (DSN), 2015 45th Annual IEEE/IFIP International Conference on*, pages 45–56. IEEE, 2015.

[124] OWASP. Threat risk modeling. https://www.owasp.org/index.php/Threat_Risk_Modeling#ST Accessed: 2018-01-15.

[125] S. Pearson and M. Casassa-Mont. Sticky policies: An approach for managing privacy across multiple parties. *Computer*, 44(9):60–68, 2011.

[126] S. Peng, Y. Yang, Z. Zhang, M. Winslett, and Y. Yu. Query optimization for differentially private data management systems. In *Data Engineering (ICDE), 2013 IEEE 29th International Conference on*, pages 1093–1104. IEEE, 2013.

[127] R. Ada Popa, C. Redfield, N. Zeldovich, and H. Balakrishnan. Cryptdb: protecting confidentiality with encrypted query processing. In *Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles*, pages 85–100. ACM, 2011.

[128] F. Prasser, F. Kohlmayer, R. Lautenschläger, and K. A. Kuhn. Arx-a comprehensive tool for anonymizing biomedical data. In *AMIA Annual Symposium Proceedings*, volume 2014, page 984. American Medical Informatics Association, 2014.

[129] A. Pretschner, M. Hilty, and D. Basin. Distributed usage control. *Communications of the ACM*, 49(9):39–44, 2006.

[130] The Privacy Office, Department of Homeland Security, Washington, DC 20528. *Privacy Impact Assessments: The Privacy Office Official Guidance*, June 2010.

[131] D. Riboni, C. Bettini, G. Civitarese, Z.H. Janjua, and R. Helaoui. Smartfaber: Recognizing fine-grained abnormal behaviors for early detection of mild cognitive impairment. *Artificial Intelligence in Medicine*, 2016.

[132] S. Royster. Working with big data. Technical report, U.S. Bureau of Labor Statistics, 2013.

[133] S. L. Salzberg. C4. 5: Programs for machine learning by j. ross quinlan. morgan kaufmann publishers, inc., 1993. *Machine Learning*, 16(3):235–240, 1994.

[134] P. Samarati. Protecting respondents' identities in microdata release. *IEEE Trans. Knowl. Data Eng.*, 13(6):1010–1027, 2001.

[135] P. Samarati and L. Sweeney. Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. Technical report, Technical report, SRI International, 1998.

[136] R. S. Sandhu and J. Park. Usage control: A vision for next generation access control. In Vladimir Gorodetsky, Leonard Popyack, and Victor Skormin, editors, *Computer Network Security*, number 2776 in Lecture Notes in Computer Science, pages 17–31. Springer Berlin Heidelberg, January 2003. 00208.

[137] SAP. Analyze sensitive data without risking data privacy. https://www.sap.com/cmp/dg/crm-xt17-ddm-data-anony/index.html. Accessed: 2018-01-15.

[138] SAP. Optimize data reliability and transparency with our sap test data migration server software and improve test management. https://www.sap.com/products/test-data-migration-server.product-capabilities.html. Accessed: 2018-01-15.

[139] SAP. Risk management and risks. https://www.sap.com/integrated-reports/2016/en/outlook-opportunities-and-risks/risk-management-and-risks.html. Accessed: 2017-12-29.

[140] A. Schmitz. Sap enterprise threat detection spot cyber attacks with speed and precision. http://www.news-sap.com/sap-enterprise-threat-detection-powerful-new-security-solution/. Accessed: 2018-01-15.

[141] Matthew A. Scholl, K. M. Stine, J. Hash, P. Bowen, L. A. Johnson, C. D. Smith, and D. I. Steinberg. Sp 800-66 rev. 1. an introductory resource guide for implementing the health insurance portability and accountability act (hipaa) security rule. Technical report, "NIST" National Institute of Standards and Technology, 2008.

[142] IBM Global Business Services. Getting smart about your workforce: Why analytics matter. Technical report, IBM CANADA, March 2009.

[143] A. Shah, S. Dahake, and S. H. Haran J. Valuing data security and privacy using cyber insurance. *SIGCAS Comput. Soc.*, 45(1):38–41, February 2015.

[144] R. A. Shaikh, K. Adi, and L. Logrippo. Dynamic risk-based decision methods for access control systems. In , volume 31, pages 447–464, 2012.

[145] A. Shostack. *Threat modeling: Designing for security.* John Wiley & Sons, 2014.

[146] J. Soria-Comas, J. Domingo-Ferrer, D. Sánchez, and S. Martínez. Enhancing data utility in differential privacy via microaggregation-based k-anonymity. *The VLDB Journal*, 23(5):771–794, 2014.

[147] L. Sweeney. Achieving k-anonymity privacy protection using generalization and suppression. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, 10(5):571–588, October 2002.

[148] M. Templ, B. Meindl, and A. Kowarik. Introduction to statistical disclosure control (sdc). *Project: Relative to the testing of SDC algorithms and provision of practical SDC, data analysis OG*, 2013.

[149] The European Commission. Data protection impact assessment template for smart grid and smart metering systems. Report, The European Commission, 2014.

[150] The European Parliament and the Council of the European Union. Regulation (EU) 2016/679 of the European parliament and of the council. *Official Journal of the European Union*, April 2016.

[151] G. Theodorakopoulos and J. S. Baras. On trust models and trust evaluation metrics for ad hoc networks. *IEEE Journal on Selected Areas in Communications*, 24(2):318–328, Feb 2006.

[152] E. Toch, C. Bettini, E. Shmueli, L. Radaelli, A. Lanzi, D Riboni, and D. Lepri. The privacy implications of cyber security systems: a technological survey. *ACM Computing Surveys (CSUR)*, x(x):xx, 2018.

[153] S. Trabelsi, A. Ecuyer, P. Cervera y Alvarez, and F. Di Cerbo. Optimizing access control performance for the cloud. In *CLOSER 2014 - Proceedings of the 4th International Conference on Cloud Computing and Services Science, Barcelona, Spain, April 3-5, 2014.*, pages 551–558, 2014.

[154] S. Trabelsi, V. Salzgeber, M. Bezzi, and G. Montagnon. Data disclosure risk evaluation. In *2009 Fourth International Conference on Risks and Security of Internet and Systems (CRiSIS 2009)*, pages 35–72, Oct 2009.

[155] S. Trabelsi, J. Sendor, and S. Reinicke. Ppl: Primelife privacy policy engine. In *Policies for Distributed Systems and Networks (POLICY), 2011 IEEE International Symposium on*, pages 184–185, June 2011.

[156] N. Ulltveit-Moe and V. A. Oleshchuk. Measuring privacy leakage for IDS rules. *CoRR*, abs/1308.5421, 2013.

[157] N. Ulltveit-Moe, V. A. Oleshchuk, and G. M. Køien. Location-aware mobile intrusion detection with enhanced privacy in a 5g context. *Wireless Personal Communications*, 57(3):317–338, 2011.

[158] J. Vaidya, C. W Clifton, and Y. M. Zhu. *Privacy preserving data mining*, volume 19. Springer Science & Business Media, 2006.

[159] S. De Capitani Di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati. Over-encryption: management of access control evolution on outsourced data. In *Proceedings of the 33rd international conference on Very large data bases*, pages 123–134. VLDB endowment, 2007.

[160] I. Wagner and D. Eckhoff. Technical privacy metrics: a systematic survey. *arXiv preprint arXiv:1512.00327*, 2015.

[161] D. Wright. The state of the art in privacy impact assessment. *Computer Law & Security Review*, 28(1):54–61, 2012.

[162] WSO2. Balana engine. `https://github.com/wso2/balana`, 2014.

[163] *XACML Obligation Profile for Healthcare Version 1.0*, February 2013. `http://docs.oasis-open.org/xacml/xspa-obl/v1.0/csd01/xspa-obl-v1.0-csd01.html`.

[164] X. Xiao, K. Yi, and Y. Tao. The hardness and approximation algorithms for l-diversity. In *Proceedings of the 13th International Conference on Extending Database Technology*, pages 135–146. ACM, 2010.

[165] H. Xu, S. Guo, and K. Chen. Building confidential and efficient query services in the cloud with rasp data perturbation. *IEEE transactions on knowledge and data engineering*, 26(2):322–335, 2014.

[166] S. Yu, C. Wang, K. Ren, and W. Lou. Achieving secure, scalable and fine-grained data access control in cloud computing. In *2010 Proceedings IEEE INFOCOM*, pages 1–9, March 2010.

[167] T. Zhu, G. Li, W. Zhou, and P. S. Yu. Differentially private data publishing and analysis: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 29(8):1619–1638, 2017.

[168] R. Zuech, T. M. Khoshgoftaar, and R. Wald. Intrusion detection and big heterogeneous data: A survey. *Journal of Big Data*, 2(1):1–41, 2015.