Risk-aware Business Process Modelling and Trusted Deployment in the Cloud

Elio Goettelmann

UNIVERSITÉ DE LORRAINE

ECOLE DOCTORALE IAEM Lorraine

Risk-aware Business Process Modelling and Trusted Deployment in the Cloud

THESIS

publicly defended the 21st October 2015

in order to get a

Doctoral Degree of Computer Science

at the University of Lorraine

by

Elio Goettelmann

Examining committee

Reviewers :	Prof. Salima Benbernou	University Paris Descartes, France
	Prof. Haralambos Mouratidis	University of Brighton, United Kingdom
Members :	Prof. Frédérique Biennier	INSA-Lyon, France
	Prof. Eric Dubois	University of Namur, Belgium
	Dr. Benjamin Gâteau	Luxembourg Institute of Science and Technology
	Prof. Laurent Vigneron	University of Lorraine, France
Supervisor :	Prof. Claude Godart	University of Lorraine, France

Fonds National de la Recherche Luxembourg

Loria Laboratoire lorrain de recherche en informatique et ses applications

LUXEMBOURG INSTITUTE OF SCIENCE AND TECHNOLOGY LIST

Layout created with the ’thesul’ class.

Abstract

Nowadays service ecosystems rely on dynamic software service chains that span over multiple organisations and providers. They provide an agile support for business applications, governments of end-users. This trend is reinforced by the Cloud based economy that allows sharing of costs and resources. However, the lack of trust in such cloud environments, that involve higher security requirements, is often seen as a braking force to the development of such services.

The objective of this thesis is to study the concepts of service orchestration and trust in the context of the Cloud. It proposes an approach which supports a trust model in order to allow the orchestration of trusted business process components on the cloud.

The contribution is threefold and consists in a method, a model and a framework. The method categorizes techniques to transform an existing business process into a risk-aware process model that takes into account security risks related to cloud environments. The model formalizes the relations and the responsibilities between the different actors of the cloud. This allows to identify the different information required to assess and quantify security risks in cloud environments. The framework is a comprehensive approach that decomposes a business process into fragments that can automatically be deployed on multiple clouds. The framework also integrates a selection algorithm that combines security information with other quality of service criteria to generate an optimized configuration.

Finally, the work is implemented in order to validate the approach. The framework is implemented in a tool. The security assessment model is also applied over an access control model. The last part presents the results of the implementation of our work on a real world use case.

Keywords: Business Process Management, Cloud Computing, Security Risk Management

Résumé

L'essor du Cloud Computing, permettant de partager les coûts et les ressources au travers de la virtualisation, présage une interconnexion dynamique et flexible entre entreprises et fournisseurs. Cependant, cette mise en commun de ressources, données et savoir-faire implique de nouvelles exigences en termes de sécurité. En effet, le manque de confiance dans les structures du Cloud est souvent vu comme un frein au développement de tels services.

L'objectif de cette thèse est d'étudier les concepts d'orchestration de services, de confiance et de gestion des risques dans le contexte du Cloud. La contribution principale est un framework permettant de déployer des processus métiers dans un environnement Cloud, en limitant les risques de sécurité liés à ce contexte.

La contribution peut être séparée en trois partie distinctes qui prennent la forme d'une méthode, d'un modèle et d'un framework. La méthode catégorise des techniques pour transformer un processus métier existant en un modèle sensibilisé (ou averti) qui prend en compte les risques de sécurité spécifiques aux environnements Cloud. Le modèle formalise les relations et les responsabilités entre les différents acteurs du Cloud. Ce qui permet d'identifier les différentes informations requises pour évaluer et quantifier les risques de sécurité des environnements Cloud. Le framework est une approche complète de décomposition de processus en fragments qui peuvent être automatiquement déployés sur plusieurs Clouds. Ce framework intègre également un algorithme de sélection qui combine les information de sécurité avec d'autres critère de qualité de service pour générer des configuration optimisées.

Finalement, les travaux sont implémentés pour démontrer la validité de l'approche. Le framework est implémenté dans un outil. Le modèle d'évaluation des risques de sécurité Cloud est également appliqué dans un contexte de contrôle d'accès. La dernière partie présente les résultats de l'implémentation de nos travaux sur un cas d'utilisation réel.

Mots-clés: Gestion des Processus Métiers, Cloud Computing, Gestion des Risques de Sécurité

Acknowledgments

Hätte man sämtliche Berge der ganzen Welt,
zusammengetragen und übereinandergestellt
und wäre zu Füßen dieses Massivs,
ein riesiges Meer, ein breites und tiefs.
Und stürzte nun, unter Donnern und Blitzen
der Berg in dieses Meer - na das würd' spritzen !
- Heinz Erhardt -

Contents

Chapter 1

Introduction

1.1 General introduction

Nowadays, modern companies have to face a lot of challenges in order to stay competitive: “innovation”, “diversification”, “upselling”, “outsourcing”, “exponential growing”, “core business refocusing”, are a few of them. While the market becomes more and more uncertain and unpredictable, such strategies aim at making a business more efficient in delivering its products or services and more flexible regarding their external dependencies (customers, partners, providers, *etc.*). Indeed, costumers want rapid and reliable services, while having fast evolving needs. Businesses must meet these expectations to keep up with huge variations in market trends, that can sometimes occur in a couple of days or weeks. In this perspective, effective information systems are not only necessary but even essential for such strategic and organizational business transformations. Hopefully, information technologies are providing means to support that, particularly through the development of “business process management systems” and “cloud computing”. Workflows can be managed in an automated and optimized way, and rely on adaptable systems that adjust to the instant needs. Therefore, companies become not only more and more automated, but also global, by depending on worldwide distributed and interrelated services.

Consequently, the loss of control over their activities, information or processes turns into a real threat due to the high interconnection of multiple, separate systems from several business domains, spread over different locations. Slight changes at any point of this complex service chain can greatly affect the overall process. Especially in cloud environments, problems like *disruption of the availability of an infrastructure*, *theft of customers personal information* or *regulations discrepancies between different jurisdictions* must be considered when outsourcing business applications. Situations can happen where the entire *know-how* and the data of a company is handled by external entities. And since the complete service chain is not under full control, the continuity of a business depends on all participants, and cannot be managed independently by one single party. Dealing with such risks is crucial: only by anticipating possible malfunctions, the service delivery can be guaranteed to respect the initial objectives. But, predictions in such complex environments are difficult to make and often depend on the trustworthiness of the different partners. And trust is a concept difficult to define through contractual agreements, standard protocols or legal frameworks. Therefore, processes must be adapted to handle uncertain situation, *i.e.* become “risk-aware”.

In this thesis we tackle the problem of adapting existing business processes to security issues encountered when deploying them into cloud environments. We adopt a risk-based approach in order to bridge the gap of uncertainty inherent to that context. This allows us to assess the security of different cloud offers and adapt the processes accordingly.

1.1.1 Context

To afford scalability and reduce investment costs, the trend is more and more to take advantage of cloud computing. Cloud computing is location agnostic and provides dynamically scalable and virtualised resources as services over the internet. Among others, it uses virtualisation, service-oriented software and grid-computing technologies. Being a distributed model, cloud computing allows accessing resources and services offered by servers from different places. Cloud computing has become a mainstream solution offering mutualisation of information technologies as a service along several paths, such as software (SaaS), platform (PaaS) and infrastructure (IaaS) [VRMCL08]. Companies such as Amazon [1], Microsoft [2], IBM [3], and Google [4], to name but a few, offer such services, which rely on virtualisation and pay-as-you-go business models. In this way, cloud computing will offer the ability to leverage IT resources, as a service, and so will make computing more accessible for all businesses [Lin09].

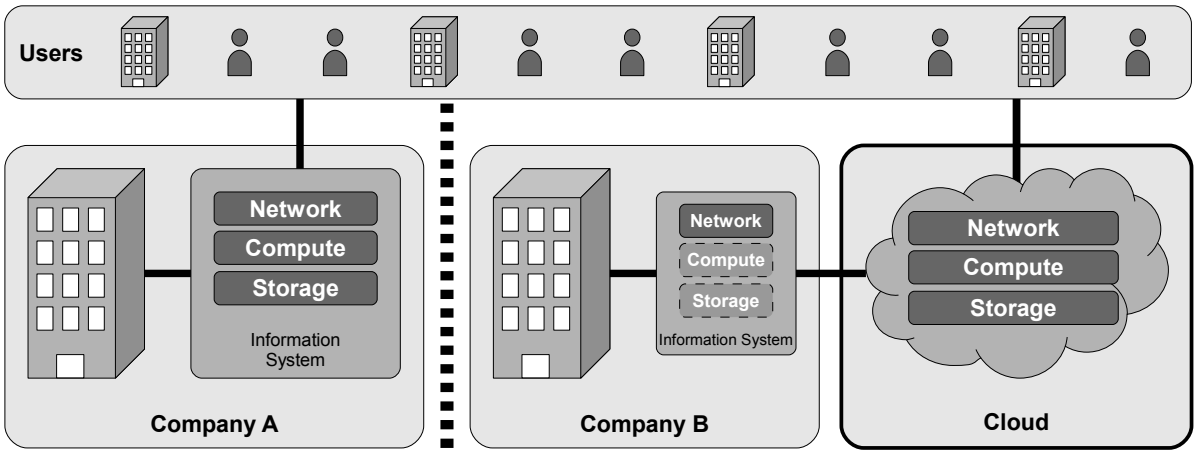


FIGURE 1.1 – Context of this thesis

FIGURE 1.1 illustrates the context of our work. It shows two companies (**Company A** and **Company B**) that rely on their respective **Informations systems** to deliver products or services to their **Users**. These users can be other companies or end-users. The difference is that **Company B** relies on cloud services to serve their customers. More detailed reasons to motivate such a decision (like cost-efficiency, scalability, *etc.*) will be given further in the manuscript. The first point to notice is that typically, a company that uses cloud services does still need an informations system on-premises ; at least some devices and network capabilities to access and manage the systems deployed on the cloud. However, this information system is substantially smaller than that of **Company A** because parts of it are outsourced on cloud services.

Our work focuses on the parts of the information system deployed on the cloud (thick border in FIGURE 1.1). Thus, we do not tackle the problems related to the information system on-premises and linked to the cloud. We argue that these security issues are common to those of a classical architecture (*cf.* **Company A**) and can be handled with existing approaches. However, externalizing essential data and know-how generates new types of threats that had not to be faced before. The architecture becomes more complex to manage, especially in terms of security (more threats, more vulnerabilities, *etc.*). This additional complexity has to be mastered to legitimize a cloud outsourcing.

1. http://aws.amazon.com/what-is-cloud-computing/
2. http://azure.microsoft.com/en-us/
3. http://www.ibm.com/cloud-computing/us/en/
4. https://cloud.google.com/

1.1.2 Motivations

Several issues are likely to emerge in the context of cloud computing. Among them is the question of multi-tenancy and the control over where the processes actually run and where the data resides. While this might not appear to be a major issue for many users at first sight, we see several reasons to actually challenge this point. The main one with regard to this thesis is the general concern over Governance, Risk and Compliance (GRC). Moreover, the problems and risks which had to be faced in Service Oriented Architecture (SOA), increase if the process executes in a cloud, where problems can come not only from the service layer, but also from the platform and infrastructure layers ; for example attacks from malicious people sharing the same virtual machine. Indeed, while some platforms start to be effective in the context of SOA architecture, enhancements are requested in the context of clouds. Since cloud computing is a new domain of research, security concerns have not been adequately addressed at all of these levels which will be the focus of the current thesis.

For these various reasons (loss of control, security, lock-in, *etc.*), widespread cloud adoption is still slowed down [5]. Moreover, recent events, such as the NSA spying scandal [6] (the PRISM surveillance program) undermined even more the already fragile confidence companies had in cloud services. Bad luck in times of crisis, where cost-efficiency should have a catalyst role for reviving economic growth. Even for the environment, adopting cloud computing would be a benefit in terms of energy savings [7]. In this perspective, re-establishing trust between cloud providers and their potential users to encourage cloud adoption, is a defensible motivation.

Accordingly, whether to trust its partners remains a basic principle for developing effective cooperation, it is nevertheless true that verifying the proper execution of transactions among different partners, clarifying doubts and establishing responsibilities in case of a conflict, are major contributions to the building of a trusted community (Trust but verify). A cloud computing infrastructure could be considered as an infrastructure involving ad-hoc networks and requiring distributed cooperation incentives schemes. Cooperation relies on trust because dependent individuals need a certain level of assurance that non-dependent individuals will not defect [MH12]. The concept of trust is abstract and difficult to define, thus several definitions exist (see [Sch10, VRMCL08]). Most of the definitions of trust come from the areas of sociology, psychology and philosophy [CRDRB09]. The definition of trust, which corresponds to our context is the one provided by D. Gambetta in [Gam88]: “Trust (or, symmetrically, distrust) is particular level of the subjective probability with which an agent assesses that another agent or group of agents will perform a particular action...”.

Therefore, measuring trust is one of the key challenges for secure cloud outsourcing, especially to enable an automated deployment of applications in cloud environments. As Brian W. Kernighan stated it: “Controlling complexity is the essence of computer programming” [KP76]. Indeed, to overcome the complexity of these new distributed and interlinked architectures, one answer is the automation of the overall outsourcing process. A better understanding of the new risks related to cloud computing will allow to identify and measure the threats. Once formalized, these risk quantifications can help to automatically adapt the processes accordingly and deploy them into cloud environments. It is in this perspective that our work should be seen, better understanding the risks of cloud computing to simplify its use through automation.

5. http://thenextweb.com/insider/2013/09/11/5-reasons-enterprises-are-frightened-of-the-cloud/
6. https://en.wikipedia.org/wiki/PRISM_(surveillance_program)
7. http://www.gogrid.com/news/2014/03/18/cloud-computing-cloud-computing-green-technology

1.2 Research problem, questions and methodology

Based on these observations and assumptions, we formulate our research problem as follows: **How to move a business application securely into a cloud environment ?** Three major points can be noticed in this research problem. First, the term **business application** limits our research to automated software for businesses. This means that we will focus on cloud services that are not intended for end-users, but for businesses that use services to deliver their own services. Typically, we deal with business-to-business relationships (B2B), driven by executable software (excluding processes that are not IT-supported). The difference is that such relationships can be formalized through business process modelling languages. Second, the term **securely** is intended to put the focus on security, meaning that it is the central matter of our research. One of our work hypothesis is that business application can be moved to the cloud under certain security conditions (at least, a loss in terms of security should be justifiable). Thus, the problem could also be seen as: **How to take into account security risks when moving a business application to the cloud ?**. The last term is *cloud environment*, which defines our context. The considered business processes must execute in a cloud environment, meaning that it is neither necessarily entirely cloud-based, nor that the process must execute on one single cloud offer. It is also intended to include the *cloud business model* and not only the technological aspects. In this sense, it means that it is important to consider security not as the only criteria motivating our work, but that it has to be combined with other criteria that drive more globally an outsourcing to the cloud.

1.2.1 Research questions

Our research problem leads us to formulating the following research questions. Answering these questions through our work, will help us to handle our research problem:

RQ.1 What are the cloud security issues ? Basically, this will bring us to focus on what is *new* when outsourcing to the cloud in comparison to a classical context. This implies defining what is a cloud environment and specifying the differences with a traditional information system, especially regarding security. In addition and to further motivate our work, answering this question will help us to draw the limits of our work: security issues of non-cloud based systems are not examined in detail in our work. This question will be answered through our state of the art in Chapter 2.

RQ.2 How to evaluate cloud security risks ? This question raises two other ones: How are these *new* risks impacting the security of the business activities ? And what information are needed from the cloud (providers) ? The objective here is to *measure* the security of a cloud-based system. If possible, in a quantifiable fashion to be able to compare different possible cloud configurations. The answer to this research question is given in one of our contributions which is detailed in Chapter 4.

RQ.3 How can cloud security risks be managed ? Indeed, there are already existing risk management approaches to deal with security issues of classical information system. But it is necessary to check if these are still adapted in a cloud context. More precisely, we will explore how these risks can be managed on business processes. An intuitive idea would be to adapt business processes to avoid/reduce these risks. The question of who can or must manage these risks is also raised. This will lead us to look at the limits of what cannot be managed on business processes. This question will be addressed in Chapter 3.

RQ.4 How to integrate the risk with other parameters ? As said previously, security has to be combined with other parameters that drive the decision of a cloud outsourcing. Cost, quality

of service or other functional requirements are often more important than security. Intuitively, our risk-based approach will help us to define a method to balance security against other parameters. But what place does the risk have among cost, QoS or other constraints, and how to automate such decisions ? This question is addressed in Chapter 5.

1.2.2 Research methodology

March & Smith [MS95] define very early the difference between *natural science* and *design science*. The first one can be seen as aiming to "understand reality", so developing concepts or languages to characterize phenomena. The second one has more the objective of "creating things that serve human purpose", so developing products.

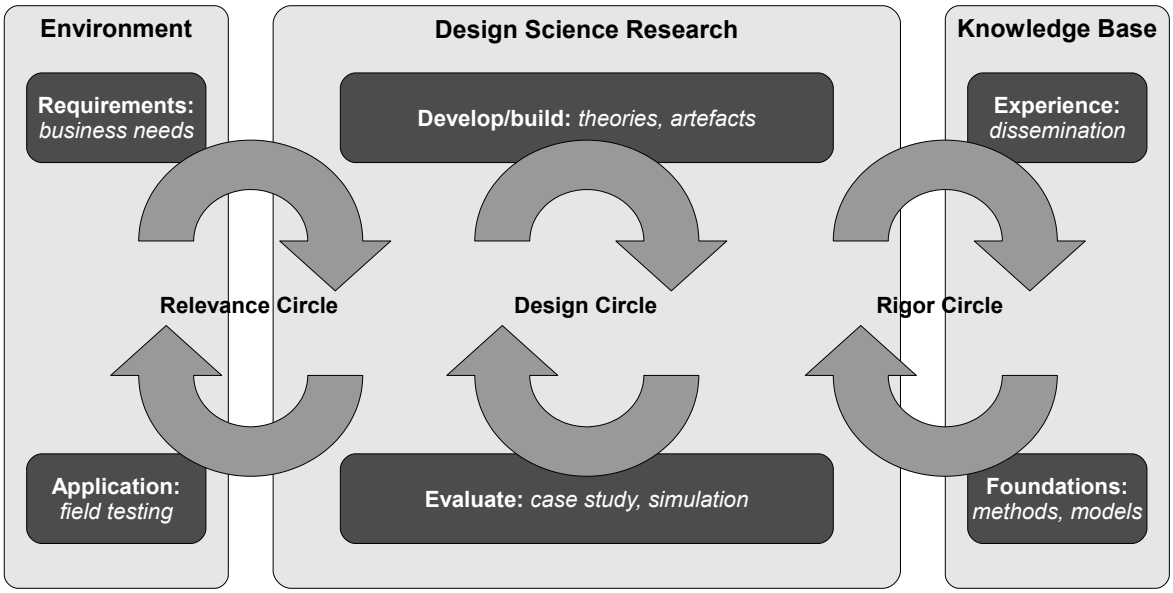


FIGURE 1.2 – Design Science framework drawn from [Wie09]

Even if our first research question could be mapped to *natural science* (we make a state of the art of cloud computing to identify its security issues), the majority of this thesis consists of *design science* (we build concepts, models, methods and tools to support the secure deployment of business processes into the cloud). Therefore, we apply a methodology based on design science research (see FIGURE 1.2). Wieringa [Wie09] defines three cycles for conducting design science research (this approach is a refinement of that of March *et al.* [MS95]):

- **The Relevance Circle** connects the design science research activities with the *environment*. It consists on the one side in providing the requirements (*business needs*) to build new tools or methods. So basically the motivations to the research work. On the other side, it brings the built artefacts and tools to the environment to apply it. This step allows to test if the developed artefacts meet the business needs.
- **The Rigor Circle** connects the design science research activities with the *knowledge base*. First, the design science builds upon existing foundations from the scientific domain. This is done with a rigorous state of the art in order to build artefacts aligned with the hypothesis of the knowledge base (*methods, models*). Second, the knowledge base is enriched, typically by disseminating the results to the scientific community through publications.

- **The Design Circle** forms the central part of design science research. It consists of building models, methods and tools based on the environment's requirements and the scientific foundations. These two inputs are meant to guarantee that these artefacts are not only relevant, but also built rigorously. Those *artefacts* are then evaluated through simulation or case studies before application and dissemination.

1.3 Summary of the contributions

Our main contributions can be narrowed down to three distinct artefacts of the **design cycle** in form of a method, a model and a framework. All three contributions have been evaluated regarding their feasibility over an illustrating example and through an implementation.

- A method for securing business processes before deploying them to the cloud (Chapter 3). It is implemented over an illustrating example in Section 3.3. This contributions answers **RQ.3**.
- A model for assessing security risks in cloud environments (Chapter 4). It is applied on an illustrating example in Section 4.3 and has been implemented in a tool in Section 6.1.1. This contribution answers **RQ.2**.
- A comprehensive framework for deploying business processes in a multi-cloud environment taking into account multiple criteria (Chapter 5). An illustrating example has been deployed through this framework in Section 5.4 and a tool has been developed to demonstrate the feasibility of the approach (Section 6.1.3). This contributions answers **RQ.4**.

To guarantee that our artefacts were built rigorously, we first carried out a state of the art that is presented in Chapter 2. This part defines our *foundations* of the **rigor circle** and answers at the same time **RQ.1**. This analysis is deepened in Section 3.1 to align the three domains of business process management, security risk management and cloud computing. The second step, the *dissemination* of our artefacts to the knowledge base, has been done in different publications:

- the method for securing business processes before deploying them to the cloud (Chapter 3) has been published as scientific papers in [GMG13] and [GMG14].
- the model for assessing security risks in cloud environments (Chapter 4) has been published as a scientific paper in [GDG+14].
- the comprehensive framework for deploying business processes in a multi-cloud environment taking into account multiple criteria (Chapter 5) has been published in [GFG13] and [GDGG14].

To identify the relevant *business needs*, the state of the art (Chapter 2) is not limited to academic publications but also includes industrial references. Through this step, the *requirements* of the **relevance circle** are identified. It helps to sustain the necessity for cloud security improvements and motivates our risk-based approach. The *application* of our artefacts to the environment is done in two ways:

- the cloud security risk assessment model has been applied over a use case during the thesis. Real cloud providers were assessed and compared to help a company to select between different available offers. The detailed case description and the results are presented in Section 6.3.
- the model has been applied in a different domain than business processes in [BGPG15] for enhancing access control. It shows that the model can be extended to other use cases of cloud computing, and thus testify of its usefulness. Details are given in Section 6.2.

1.4 Running example

In order to illustrate the contributions of this thesis, we introduce a running example which will be used all along the manuscript. It will help to illustrate the different concepts, approaches and techniques defined in our work to better understand their mechanics and their consequences.

This example describes a company which wants to outsource some parts of their IT to the cloud.

1.4.1 Description of the company

The company is a (fictitious) firm which sells different products to customers located in Europe. It has basically an online shop providing a catalogue of their products that potential customers can access and buy. The company has a significant customer database and a great experience with online sales as it became active 15 years ago. Today the company faces a competitiveness problem since many other actors emerged with high-availability websites and very fast-shipping services. To focus on their core business, *marketing, packaging and shipping their products*, the company wants to consider emerging cloud technologies. The company also hopes that the outsourcing of their infrastructure and business applications will reduce their operational costs. Moreover, it will probably help to target worldwide customers as their current infrastructure is not scalable enough to support rapidly such an evolution. However, this potential cloud outsourcing raises some security issues that the company wants to examine. Indeed, the company does not want to unnecessarily expose its know-how and its sensitive data. News about data-breaches and exposure of personal information are currently very common, and those risks have to be taken into account. Moreover, the operational costs of running their applications on a cloud infrastructure have to be analysed in detail to be sure that costs will be actually reduced.

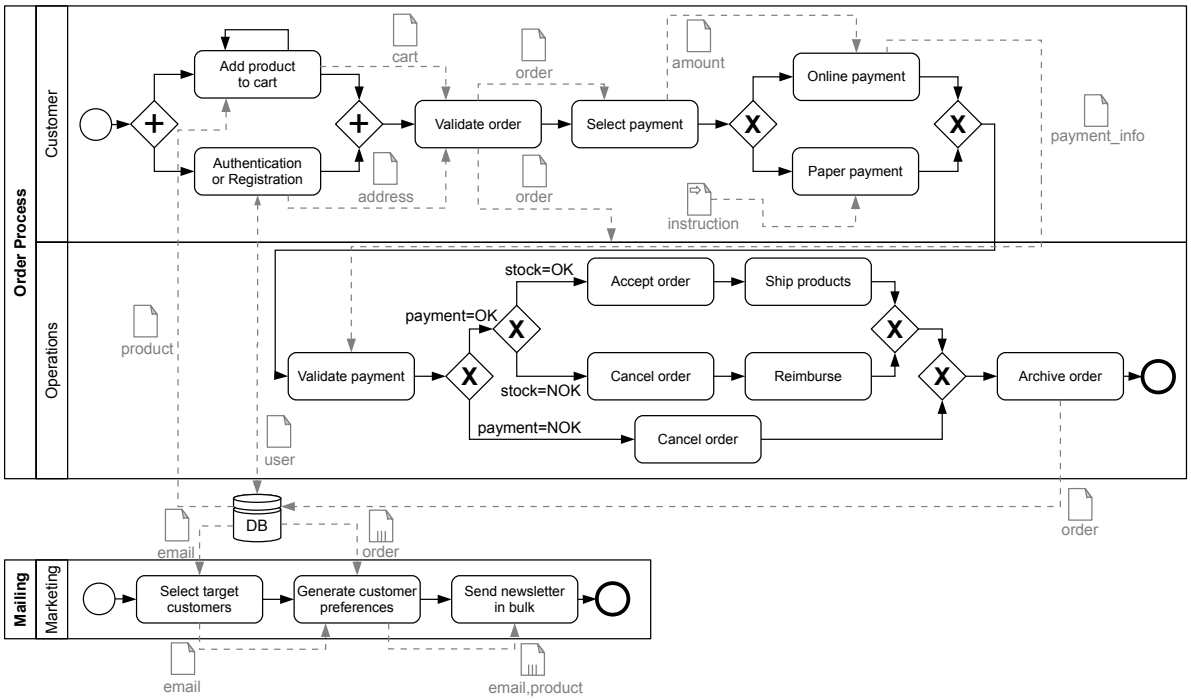


FIGURE 1.3 – Illustrating example - Business processes candidate for being outsourced

The applications candidate for being outsourced can be summarized in two business processes illustrated in FIGURE 1.3 using the BPMN 2.0 notation.

The first one is the **Order Process** which corresponds to the handling of an order. It is decomposed into two *swim lanes*, one for the actions of the *Customer*, the other one for the actions performed by the company's *Operations Department*. The process starts as soon as a customer accesses the website. He can add the available *products* to a *cart* through the repeatable **Add product to cart** task. Once his purchases finished, he can **Validate the order** after he has logged in (or registered if he hasn't any account yet) through the **Authentication or registration** task. The *cart* is transformed into an *order* by adding the user's *address*. There are two available payment options, either the customer enters his *payment_info* on an online form, or the customer can decide to make a bank transfer. In the second case, instructions are given where to transfer the payment. At this point an authorized *operation manager* can **Validate the payment** once the entered payment information are verified. If the payment cannot be verified, he can **Cancel the order**. Similarly, if the current stock does no longer permit to **Ship the products**, the *order* is **Cancelled**. In any case, the order is **Archived** to statistically analyse the sales of the company.

The **Mailing** process uses these information to send suggestions to the customers. A *marketing manager* can regularly send newsletters to **Selected target customers**. The company has developed an algorithm which can **Generate customer preferences** based on the order archive. Thus, a personalized email can be created with a list of *products* that may interest the customer. The generated newsletters are sent to the customers in bulk through an internal mailing system (**Send newsletter in bulk** task).

Data objects (like *product*, *cart* or *order*) are represented with dashed grey arrows. They mean that the data element is *produced* by the first task and *available* for the second one. For example, the *cart* element is created by the **Add product to cart** task and is used by the **Validate order** task to create an *order*. To note is that the **Validate order** task sends the *order* object to the entire *Operations* swim lane, this means that this data object is accessible by all tasks of this swim lane. The annotations in black (like *payment=OK* or *stock=NOK*) are conditions for the gateways that specify which branch is executed. An object persistence system is visible in form of a database (**DB**) between the two business processes. More details about the used notation can be found on the BPMN 2.0 poster [8] published by the BPM Offensive Berlin.

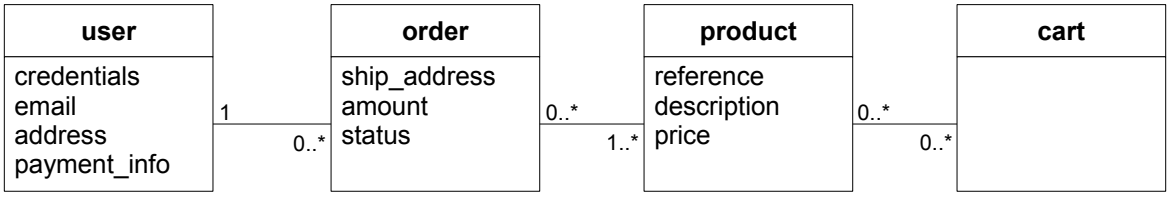


FIGURE 1.4 – Data objects of the business processes

A simplified UML class diagram is presented in FIGURE 1.4. It shows the different data objects and their attributes needed by these two processes. Note that the *cart* object does not hold any information, since it is only a container for the products that the customer is buying (the cart is never persisted). The detailed definition of those data objects will help to evaluate the security risks related to the execution of the processes in a cloud context. Indeed, different data objects or attributes can have different security requirements.

8. `http://www.bpmb.de/images/BPMN2_0_Poster_EN.pdf`

1.5 Structure of the manuscript

In addition to the introduction, the manuscript is organized in five main chapters:

- **Chapter 2:** is the state of the art of our work. It is categorized in three distinct parts, one for cloud computing, one for security risk management and one for business process management.
- **Chapter 3:** is the first contribution which is the method for securing business processes before deploying them to the cloud. The first section of this chapter deepens the state of the art of the three studied domain to build a comprehensive model for our approach.
- **Chapter 4:** is the second contribution which is the cloud security risk assessment model. It formalizes the different concepts of our approach and describes how to quantify security in cloud environments.
- **Chapter 5:** is the third contribution which is the framework for decomposing business processes over multiple clouds by taking into account other criteria in addition to the risk level.
- **Chapter 6:** is the implementation and validation of the three contribution and is separated in three parts. The first part concerns the tools developed during this thesis. The second presents the instantiation of our assessment model in the domain of access control. And the last one is a real world use case.

The last chapter presents the perspectives of our work and possible follow ups, its limitations and finally concludes the manuscript.

Chapter 2

State of the art

Structure

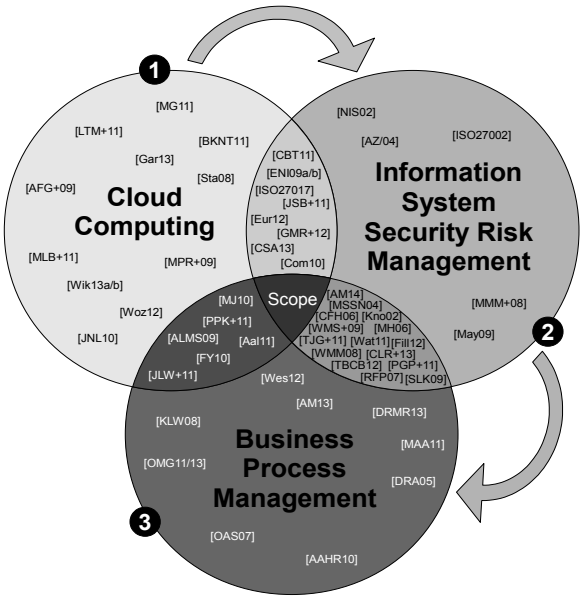


FIGURE 2.1 – Structure of the state of the art and scope of the thesis

This thesis is at the intersection of three research fields: Cloud Computing, Information System Security Risk Management and Business Process Management. It is this intersection which makes the thesis an original work and allows it to stand out from existing contributions. In the following state of the art, each of these different fields will be tackled, as their respective intersections. It will be organized as shown in FIGURE 2.1. First we will define **Cloud Computing** (Section 2.1) and give an overview of the remaining challenges of this research field (Section 2.1.2). Then we will expose **Information System Security Risk Management** (Section 2.2) and how this domain can help to solve some of the issues from the Cloud Computing domain (Section 2.2.2). Finally, we will introduce **Business Process Management** (Section 2.3) and link it to the two previous presented research fields.

2.1 Cloud Computing

Historically, cloud computing has its origins in the 50's, when the first computers, accessible through remote terminals, made their apparition (mainframes). The idea was to share computational resources to use them in an optimized way [Wik13a]. Currently, cloud computing can be seen as the final outcome of the different evolutions in technology of the last years [BKNT11]. The convergence of the technological advances like the internet, optical fibre and multi-tenancy/virtualisation, as the definition of several standards, enables today to perceive information technology as a unique entity, accessible on-demand, as a self-service, shared and configurable. From a pure hardware point of view, cloud computing could be outlined as *the illusion of infinite computing resources* [AFG+09].
The expression itself, *cloud*, comes from the common modelling practices in networking, where the internet is often represented as a stylized cloud [Wik13a].

2.1.1 Definition

According to what was previously defined, it could seem that cloud computing is nothing new after all. Although, regularly such type of talk can be found, as it is implied on the French version of the Wikipedia article about cloud computing:

> *"The cloud computing model renews well-known notions of information technology (the notion of service, multi-tenancy/sharing, virtualisation, ...) with a new commercial and marketing speech (and especially many Anglicisms)." [Wik13b]*

Indeed, it can be considered that cloud is not a technological advance, since in terms of technology it brings nothing new. Whereas, restricting the concept of cloud to a commercial speech is very limiting. What the cloud announces is revolutionary in the sense that it proposes to redefine how information technology is used. As Bitkom suggests it in the title of its white paper [MPR+09], which summarizes the situation in a very simple way: "Cloud computing - Evolution in technology, revolution in business". Thereby, Bitkom suggests following definition:

> *Cloud computing is a form of demand-driven and flexible use of IT performance. These are made available in real time as a service over the Internet and billed according to use. Thus, cloud computing allows users to transform investment expenses into operating expenses. [MPR+09]*

This definition shows more clearly what cloud computing really brings: the novelty lies in the completely different manner of using information technology. Often, there is made a parallel between cloud and electricity, which allows to give clear examples. Electricity suppliers have completely replaced the need of on-premise power production. They are in charge of producing and distributing electricity and bill their customers according to the consumed amount (as several other parameters like phasing). So cloud computing could be assimilated to *digital energy*, generated in a centralized way and provided on demand. So the novelty resides less at the level of the provider (who produces the *energy* in the same way as before, but at greater scale), but more at the level of the consumer (who has to use it in a completely different way).
In this sense, a list of essential characteristics of cloud computing can be made [MG11], [AFG+09]:

- **On-demand self-service** - resources are available as requested from the consumer, automatically, without human interaction or re-negotiation with the provider.
- **Broad network access** - to access the different services an internet connection is required (or at least a network access in the case of a local cloud).

- **Resource pooling** - or virtualisation, the physical resources are shared among multiple consumers in order to assign and reassign them dynamically.
- **Rapid elasticity** - resources can be scaled rapidly (and automatically) according to the consumers need, who can consider them as infinite.
- **Measured services** - or pas-as-you-go, the consumer pays only what he uses, unlike before, where he had to invest in equipment and thus predict his needs.

This last point is the main argument of cloud computing, it transforms CAPEX (capital expenditures) into OPEX (operational expenses): the information system does no longer require investments into infrastructure or software, it is billed according to the use. Therefore, companies can become more flexible. Moreover, it is expected that resources will become less expensive in this way of use. By working at a greater scale, a supplier can better optimize the energy consumption (this time electrical energy) than in today's decentralized utilization. This is also an important argument for the current environmental issues: reducing energy consumption is part of the answer to global warming. Although, sometimes the term of *green computing* can be heard in combination with cloud computing [9].

So, basically cloud computing could be assimilated to *digital energy* and will have its major impact on the way we use information technology. In the following we will expose more clearly what this *digital energy* is and use the term of *Cloud Services* instead. To better explain what are the kinds of services available through cloud computing, we will first present the cloud architecture.

2.1.1.1 The cloud architecture - The three service levels

The cloud architecture can be separated in three levels (see FIGURE 2.2), which represent the types of services accessible through cloud computing [BKNT11]. This separation is not strict, since some services can be assigned to more than one of these levels, while other do not match any of the three.

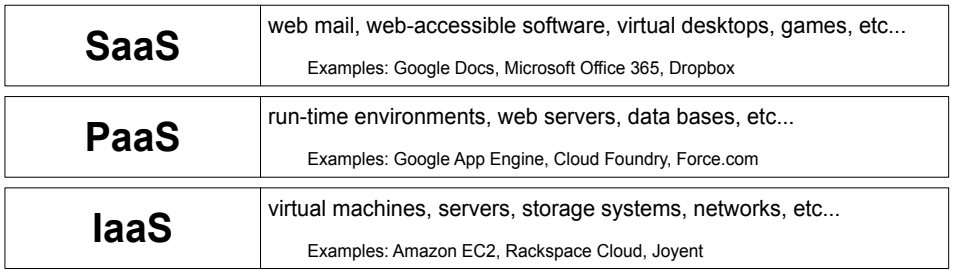


FIGURE 2.2 – The three service levels of cloud computing

Infrastructure as a Service (IaaS) It is the lowest level, and thus also the most developed one until today. The principle consists in providing *physical* resources on-demand (as a service). Through an interface, the user can manage the provided resources such as: servers, virtual machines, storage systems, networks, *etc.*. Thereby, the user is able for example to install/remove operating system

9. `http://www.greenbiz.com/blog/2011/07/27/4-reasons-why-cloud-computing-also-green-solution` (2013)

(OS) images, start/stop OS instances or even change characteristics of virtual machines (add a core, add storage space, add memory, *etc.*).
Examples of IaaS providers are: Amazon EC2, Rackspace Cloud, Joyent

Platform as a Service (PaaS) In this second level of service, typically the operating system is already provided. This level addresses essentially developers by providing a platform on which they can: develop (development environments), execute (execution environments) or test (testing environment). This level also includes all offerings of database management systems (DBMS) and web servers. Business Process Execution Engines, presented in Section 2.3, when offered as a cloud solution, would typically enter in this category of services. So, at this level, the provider is accountable for the physical resources, the user is not concerned with their management. This level of services is set on top of an IaaS, which can be provided by the same provider or another one.
Examples of PaaS providers are: Google App Engine, Cloud Foundry, Force.com

Software as a Service (SaaS) This is the highest level, which provides a software to the final user through, most of the time, a web interface. The main benefit is that there is no installation required on the client side, maintenance, update and support are clearly simplified in this way. Another interesting aspect is that the application is independent of the terminal/device used by the consumer, only an internet connection is required. This is helpful in the growing mobile world. Instead of buying the license of a software, installing it on the company's computers and running it from there as in a classic information system, an account is created and the software is directly accessible through a web browser. The price can then depend of the number of created accounts, the storage space used, the functionalities used and so on.
Examples of SaaS providers/offerings are numerous, because at this level any type of application can be found, for the end user, up to dedicated software for some companies. Nevertheless, the well-known are: Google Docs, Microsoft Office 365, Dropbox

Other types of services Other types of services can be found, which are not necessarily specific to the cloud, but whose expansion is bound to the adoption of cloud computing. An example is Data as a Service (**DaaS**), which consists in providing data on demand [Wik13b], as several governments are doing it through the Open Data initiatives, to promote, among other things, transparency of administrations. Another example is Human as a Service (**HuaaS**), which enlightens that the concept of cloud computing is not limited to information technology [BKNT11]. This type of service considers humans as a resource to accomplish specific tasks. *Crowd-sourcing* is derived from this concept, and uses the capacities of groups of humans to realize activities which can not be automated. The platform *Mechanical Turk* [10] from Amazon is a precursor in this domain. Other types of services are **BPaaS** (Business Process as a Service), **StaaS** (Storage as a Service), **NaaS** (Network as a Service), which are generally aggregated into XaaS (X as a Service).

2.1.1.2 The deployment models

Generally, three deployment models are considered for the cloud (*cf.* FIGURE 2.3), even if others can be added [Sch10].

- **public**, this type of cloud is accessible to everybody (for free or not). In most of the cases it is an outsourced service offering by one provider, who is proposing the same offering to other users. In principle, these types of offering are the most interesting in terms of financial costs.

10. `https://www.mturk.com/mturk/` (2013)

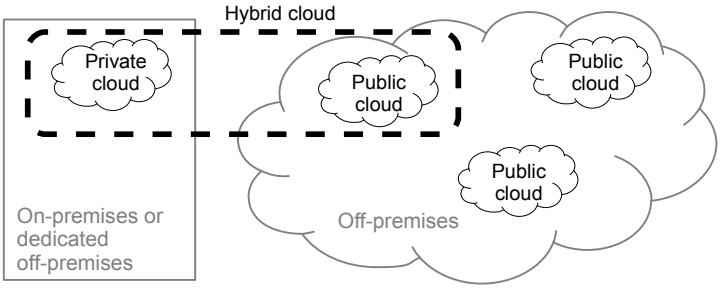


FIGURE 2.3 – Cloud deployment models

- **private**, is a cloud dedicated to one user/company/federation. It can be local/in-house, managed by the user itself (internal to the company), or outsourced. In principle, these offerings are the most flexible in terms of configuration possibilities, as it is dedicated, it can be exactly adapted to the needs.
- **hybrid**, proposes to combine the two previous models by deploying one portion of an application on one or more public clouds, and the other portions on one or more private clouds. So, this type of clouds offers the advantages (and disadvantages) of both models.

Even if the cloud rather proposes to centralize resources and applications, **community** clouds can also be cited, which propose to aggregate resources/services of several different entities, to offer one cloud solution but which will be physically distributed (in opposition to a classical data-centre). A well known example of such a configuration is the *Folding@Home* project [11], where unused power of personal computers were used in medical research to determine the mechanisms of protein folding.

2.1.1.3 The cloud actors

Different actors can be identified within the cloud computing model. In fact it can be considered as slightly more complex than the classic client/server interaction. There are different studies which tried to identify and define the different stakeholders of cloud computing [LTM+11], [MLB+11]. In the context of our work, three major actors can be identified, and eventually a fourth. The others will be discussed without any further explanations.

Cloud Consumer Is defined by the National Institute of Standards and Technology (NIST) [LTM+11] as "a person or organization that maintains a business relationship with, and uses service from *Cloud Providers*". He could also be denoted as client, customer, subscriber or user. The cloud consumer subscribes to a service proposed by a provider, uses it according to its needs and may have to pay for it if the delivered service is not free. Thus, a cloud consumer can be an end-user, a company or even a cloud provider which is relying on another cloud service for providing its services.

Cloud Provider Is "a person, organization, or entity responsible for making a service available to interested parties" according to the NIST [LTM+11]. He can also be denoted as supplier. The provider offers and delivers services to the cloud consumer. He may be the owner of the computing infrastructure, but he can also be using it from another provider to deliver its own upper level service. Thereby a cloud platform provider, can be the cloud consumer of a cloud infrastructure provider. *Dropbox* for example uses the cloud storage system of *Amazon* to deliver its services [12].

11. `http://folding.stanford.edu/home/`
12. `https://www.dropbox.com/help/7/en`

Cloud Broker The NIST defines the cloud broker as following [LTM+11]: "an entity that manages the use, performance and delivery of cloud services, and negotiates relationships between Cloud Providers and Cloud Consumers". A cloud consumer can subscribe to a service offered by a cloud broker, instead of contacting directly a provider. This definition proposes to categorize the broker services into three types:

- *Service Intermediation* - In this case, the broker enhances an offering of a provider by adding a specific layer (like reporting).
- *Service Aggregation* - By combining multiple services into new services which are more interesting or more adapted to the cloud consumer's needs.
- *Service Arbitrage* - By comparing different cloud offerings, the broker can propose to select the most appropriate provider (this can be done automatically).

Other actors Different other stakeholders of cloud computing can be defined. A **Cloud Auditor** for example is responsible of evaluating cloud services in an independent way to produce valuable information about security, privacy, performance, *etc.*. A **Cloud Carrier** provides the transport layer of the cloud services, so basically the network which connects the consumer to the provider. Governments or other legal entities have also an important role for enabling cloud computing within an adequate regulatory environment, these actors can be regrouped in a **Cloud Regulator** category. Manufacturers or software editors take also an important place in the development of cloud computing by providing facilitating hardware and software, these can be called **Cloud Enablers**.

2.1.2 Cloud challenges

Cloud computing is still emerging and is far from reaching its full potential. According to Gartner [Gar13], from 2013 to 2016 about $677 billion dollars will be spent on cloud services worldwide. But already a lot of warnings about cloud issues are upcoming. GNU creator and founder of the Free Software Foundation, Richard Stallman, called cloud computing "a trap" because user would "lose control" [Sta08]. Apple co-founder Steve Wozniak feels the same:

> *"[...] there are going to be a lot of horrible problems in the next five years. [...] With the cloud, you don't own anything [...] the more we transfer everything onto the web, onto the cloud, the less we're going to have control over it." [Woz12]*

Indeed, one of the major problems in a cloud context is that the information system is no longer under full control, and thus the consumer is relying on the architecture defined by the provider. It is not possible to constrain a remote cloud as easily as an infrastructure on-premises. Therefore, before adopting cloud computing at larger scale, it would be good to be a bit more suspicious about cloud and to know the kind of issues the adopters will have to face. To answer these critics, different research studies have been done, in order to give an overview of the remaining challenges of cloud computing [GMR+12],[AFG+09],[CBT11]. Marston *et al.* [MLB+11] proposes a SWOT analysis in which they identify several threats like the lack of standards, the different regulation levels (local, national and international) and of course, security. In these sense, the Cloud Security Alliance (CSA) made an interesting list of top threats to cloud computing [CSA13]:

- **Data Breaches**, when the organization's sensitive internal data falls into the hands of their competitors. Design flaws of a service could allow attackers to access the client's data.

- **Data Loss**, when an attacker, a physical catastrophe or another incident leads to the permanent loss of customer's data. When data is encrypted, the loss of the encryption key would have the same results.
- **Account or Service Traffic Hijacking**, when an attacker gains access to the user's credentials and can reuse them for another purpose. Typically this can be achieved through phishing, fraud or Cross-Site Scripting (XSS).
- **Insecure Interfaces and APIs**, when the interfaces provided by a cloud service presents vulnerabilities, these can lead to accidental or malicious attempts to circumvent policies.
- **Denial of Service**, when the system is slowed down to a point that the service is no longer accessible. Often this is achieved by overloading the service with many requests (Distributed Denial-of-Service attack), as it was done against Spamhaus in 2013 and slowed down a big part of the internet [13].
- **Malicious insiders**, when a current (or former) employee has an authorized access and uses it to negatively affect the system or its information. A good example is the theft of 77 million accounts of Sony's PlayStation Network [14].
- **Abuse of Cloud Services**, when the service is used for a malicious purpose, like performing DDoS attacks, cracking encryption keys or distributing pirated software. This could lead to the service shut-down and affect other consumers (like the Megaupload shut-down in 2012 [15]).
- **Insufficient Due Diligence**, when the risks of using cloud services are not completely understood by the consumer. It is possible that the cloud service does not fully meet the consumer's requirements, and thus expose him to new kinds of threats (as the 2014 celebrity photo leaks [16]).
- **Shared Technology Vulnerabilities**, when a vulnerability or a misconfiguration of any of the delivery layers can lead to the compromise of the entire stack. Multi-tenant architectures have to offer a strong isolation between each component. The best known example of that is the Blue Pill designed by Joanna Rutkowska [17]

One of the most meaningful events showing the importance of those threats was the 2013 NSA spying scandal, which heavily impacted businesses of the cloud industry [18]. Through PRISM, the mass electronic surveillance data mining program, the NSA could collect personal or business information, even of non-US entities. It resulted in an important loss of trust in the major Internet companies such as Google, Microsoft or Facebook, which were accused of participating on a voluntary basis to such a program. This event also underscored the dynamism of the cloud context, since this kind of threats were not really taken into account before mid-2014 and it did for example not appear in the CSA list of top threats to cloud computing in 2013 [CSA13].

All these challenges underscore the need for novel approaches, methods and tools to manage the security of these new kinds of information systems deployed on the cloud. Thus, these challenges

13. http://blog.cloudflare.com/the-ddos-that-almost-broke-the-internet/
14. http://en.wikipedia.org/wiki/PlayStation_Network_outage
15. http://www.bbc.com/news/technology-16642369
16. http://en.wikipedia.org/wiki/2014_celebrity_photo_leaks
17. http://en.wikipedia.org/wiki/Blue_Pill_(software)
18. http://business.time.com/2013/12/10/nsa-spying-scandal-could-cost-u-s-tech-giants-billions/

can often be found in funding programs for research activities [19, 20] and are the main motivations for this work.

19. http://ec.europa.eu/programmes/horizon2020/en/h2020-section/information-and-communication-technologies
20. http://ec.europa.eu/programmes/horizon2020/en/h2020-section/secure-societies-%E2%80%93-protecting-freedom-and-security-europe-and-its-citizens

2.2 Information System Security Risk Management

Managing risks on information systems is essential for companies to guarantee their security while handling costs. Different standards and methodologies exists to support companies in this task depending on their context. Generally, the risk management process consists in identification, evaluation and prioritization of the risks before taking actions to reduce or prevent them. The basic idea behind security risk management is to accept the fact that implementing a completely secure system is either impossible or too expensive to be a realistic solution. Thus, the risk management process tries to classify by importance the risks which could adversely affect the information system and treat them to achieve an *acceptable level of risk*. With such an approach it becomes possible to *measure* security in a qualitative manner but also in a quantitative manner through financial costs. In some areas risks can also be considered having a positive effect, such as in financial markets. But in our context we limit risks to a potential negative effect on an information system.

2.2.1 Definitions

Roughly speaking, a risk is defined as the combination of the probability that an event occurs and its consequences [NIS02]. In the IT security context, where IT components (*e.g.*, hardware, network, *etc.*) support business assets (*e.g.*, information, processes, *etc.*), the security risk is defined in a more fine-grained fashion. The event is usually seen as a threat which uses one or more vulnerabilities of the IT components in order to create a negative impact (*e.g.*, destruction, alteration, theft, *etc.*) on the business assets [May09]. For instance, an attacker steals customers' information (*i.e.* **threat**) through a compromised interface (*i.e.* **vulnerability**) which leads to the business reputation loss (*i.e.* **impact**).

2.2.1.1 Vocabulary - The ISSRM domain model

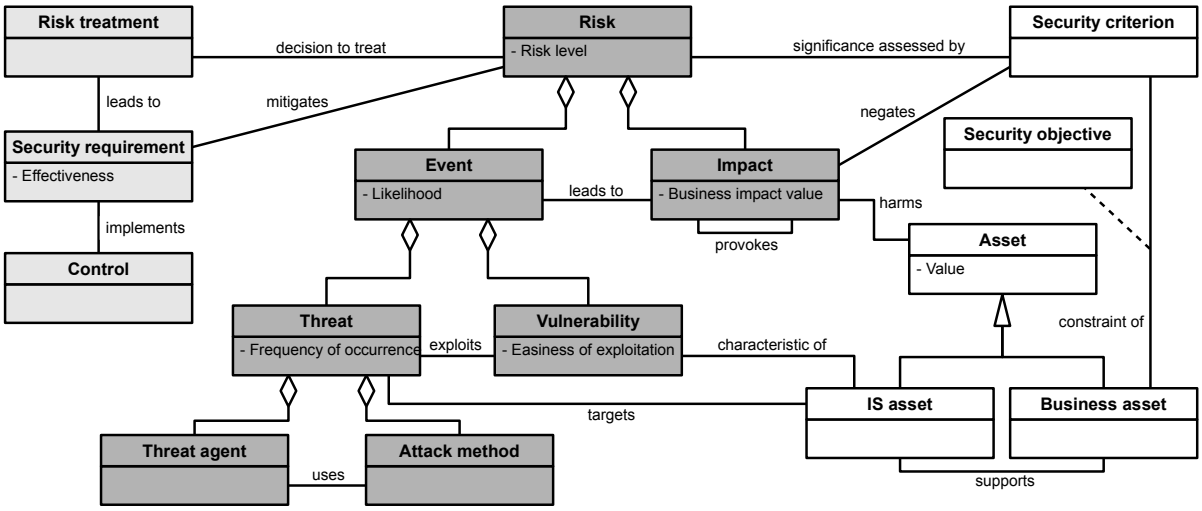


FIGURE 2.4 – ISSRM Domain Model

In this sense, a security risk assessment consists usually in evaluating the following formula ([AZ/04], [NIS02]):

$$Risk = Vulnerability \times Threat \times Impact \tag{2.1}$$

The goal is to estimate security risks in a qualitative and/or quantitative manner, to select those that need to be reduced and to develop countermeasures. Developing countermeasures involves the implementation of security controls by restraining technical solutions and by reducing vulnerabilities on the business settings. Security controls are management, operational, or technical safeguards prescribed for an information system to protect the confidentiality, integrity, and availability of the system and its information. Examples are *firewalls* or *intrusion detection systems*.

FIGURE 2.4 shows the ISSRM domain model presented by Mayer *et al.* [MMM+08]. This model integrates the main concepts related to the domain of Information System Security Risk Management. Basically it describes what a security risk consists of, what it affects and how it can be countered.

In this approach we can notice that the impact of a risk is given by the affected business asset through the definition of a **security criterion**. Security criteria are often defined over the three CIA security attributes: *Confidentiality*, *Integrity* and *Availability* [NIS02]. Sometimes other attributes can be added such as *Legality*, *Robustness*, *Minimal performance* or *Scalability*. On the other side, a **risk treatment** can counter the risk through the implementation of **security controls**.

In the following we will detail the common risk management process to achieve the definition of secure information systems.

2.2.1.2 The common security risk management process

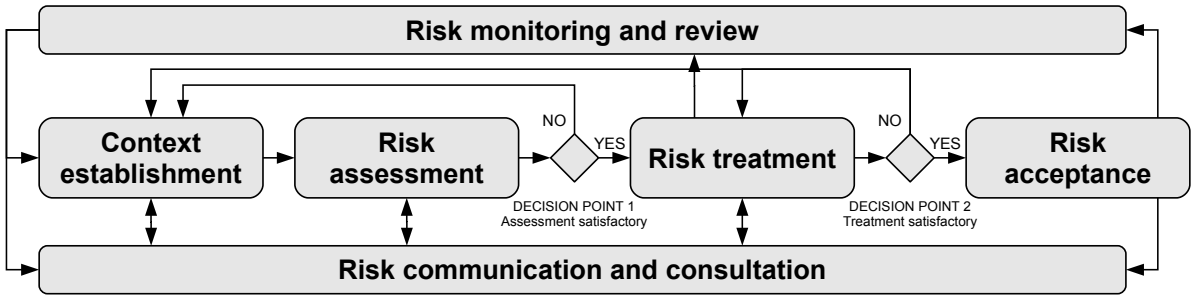


FIGURE 2.5 – Common Risk Management Process

The most common and standard risk management process depicted in FIGURE 2.5 involves the following activities [ISO11]:

- **Establishing the context** of the organization, including the definition of the scope, objectives and context of the risk management process and making clear what criteria will be used to evaluate the significance of risk.
- **Assessing the risks**, that means identifying sources of risk and areas of impacts, analysing the risks through the estimation of the consequences of risks and the likelihood that those consequences can occur, and finally evaluating which risks need treatment and their priority level.
- **Treating the risks** via the selection of risk treatment options and definition of risk treatment plans. The risks are then assessed again to determine the residual risks: risk remaining after risk treatment.
- **Accepting the risk** treatment plan and the residual risks by the organization's managers.

In parallel of the preceding activities, it is also necessary to regularly **monitor and review** the risks and the underlying risk management process. Moreover, **communication and consultation**

with the different stakeholders should take place during all stages of the risk management process.

2.2.1.3 Security risk treatment strategies

Concerning the risk treatment, information security risk management suggests four different strategies defined as follows [ISO11]:

Definition 1 (Risk modification) *The level of risk should be managed by introducing, removing or altering controls so that the residual risk can be reassessed as being acceptable.*

Risk modification involves improvement of the information system, leading to an increase of the information security level. Typically, technical safeguards or countermeasures are put in place to either reduce the vulnerabilities or reduce the consequences of an attack. It can sometimes also be called risk reduction strategy.

Definition 2 (Risk retention) *The decision on retaining the risk without further action should be taken depending on risk evaluation.*

The risk is accepted as it is, but this decision is informed and the accepted risk is subject to monitoring and review. Typically, a risk can be accepted when its level is below a previously defined *level of acceptable risk*, but sometimes it is impossible (or too expensive) to take any actions against a risk, and there is no other option than accepting it.

Definition 3 (Risk avoidance) *The activity or condition that gives evidence that the particular risk should be avoided.*

In this case, modifications occur before implementing the information system of the organization in such a wise that the risk no longer can occur. It is not a technical solution such as for *risk modification* but rather an architectural or even business decision. For example, in the context of a cloud outsourcing, a company could assess some cloud risks as being too high and thus take the decision of not outsourcing some parts of its IS to avoid those risks.

Definition 4 (Risk sharing) *The risk should be shared with another party that can most effectively manage the particular risk depending on risk evaluation.*

The most common example for this case is when a company takes an insurance for covering some risks (such as fire). It is important to note that in the case of risk sharing, new actors will be involved. Risk sharing is particularly relevant in the context of cloud computing where some processes, or parts of them, can be outsourced.

2.2.2 Security risk management in the context of cloud computing

According to the previous presented elements, security risk management seems to be an adapted approach to increase the security of cloud services and applications. It gives tools and methods to decision makers to balance security against costs by abstracting the technical details of security. Basically it should go hand in hand with cloud computing, which goal is to abstract the technical details of the information system.

2.2.2.1 Evaluation of cloud security risks (ENISA)

The most complete study to this point is the risk assessment accomplished by the European Network and Information Security Agency (ENISA) [ENI09a]. This report gives a list of 35 risks related to the use of cloud computing, and sorts them into 4 main categories. The risks are also related to assets and vulnerabilities, which allows to quantify the risk in terms of impact and probability and classify the risks following three levels: Low, Medium and High. Since it is not applied on a real use case, the values are rather informal and cannot be taken as universal values.

Policy and organizational risks These are the risks coming from the strategy for deploying an application on the cloud or directly from the structure of cloud computing in general. Thus, it is difficult to implement security controls to reduce those risks. They could be reduced in general through the adoption of standards or regulations defined at a broader level.

- **High** level of risk
  - **R1. Lock-in**: when switching costs are too high, the customer is unable to use another solution and becomes dependent on a specific provider.
  - **R2. Loss of governance**: changes in the terms and conditions of a service may lead to a loss of compliance to the security requirements.
  - **R3. Compliance challenges**: a cloud provider may be not able to comply with some requirements of the customer (standards or regulations).
- **Medium** level of risk
  - **R4. Loss of Business reputation due to co-tenant activities**: malicious activities performed by one customer may affect the reputation of another.
  - **R5. Cloud service termination or failure**: a provider could go out of business or change its services, which would directly affect the customer.
  - **R6. Cloud provider acquisition**: changes in the strategy of the provider could create some compliance issues.
  - **R7. Supply chain failure**: if the provider relies on third parties, this creates critical dependencies (the toughness of a chain always depends on the weakest link).

Technical risks These are the risks coming from the technologies supporting cloud services and are depending of technical choices made by the cloud provider. Therefore, these risks can be reduced by the implementation of technical countermeasures, but often they can only be implement by the cloud provider itself. The values of these risks are completely dependent of the selected cloud provider or service, as some providers may be more secure than others.

- **High** level of risk
  - **R9. Isolation failure**: failures of mechanisms separating cloud computing environments can lead to loss of sensitive data, reputation damage or service interruption.
  - **R10. Cloud provider malicious insider - abuse of high privilege roles**: a corrupt employee for example could threaten the confidentiality, integrity or availability of data.
- **Medium** level of risk

- **R8. Ressource exhaustion (under or over provisioning)**: the main consequences would be service unavailability and reputation loss.
- **R11. Management interface compromise (manipulation, availability of infrastructure)**: accessible through an internet browser, these interfaces present more vulnerabilities.
- **R12. Intercepting data in transit**: not all connection are secure, and even then, man-in-the-middle attacks, sniffing, spoofing, side channel and replay attacks are still existing threats.
- **R13. Data leakage on up/download, intra-cloud**: this risk is specific for the link between the cloud customer and the cloud provider.
- **R14. Insecure or ineffective deletion of data**: beyond the lifetime specified in the security policy, data may be available to other parties which may use it in a malicious way.
- **R15. Distributed denial of service (DDoS)**: a well-known issue, even more important when more business logic is web-accessible.
- **R16. Economic denial of service (EDoS)**: not the availability of the service, but the customer itself is targeted to damage him economically.
- **R17. Loss of encryption keys**: if malicious third parties get access to passwords or secret keys the whole customers business is threatened.
- **R18. Undertaking malicious probes or scans**: even if not gaining access to the assets, an attacker could get other interesting information a business.
- **R19. Compromise service engine**: a known existing vulnerability in the virtualisation layer for example could be exploited and would easily damage a customer.
- **R20. Conflicts between customer hardening procedures and cloud environment**: two customers could have conflicting security requirements and they have to know how the provider deals with such issues.

Legal risks These are the risks coming from insufficient coverage of regulations concerning cloud services or specific laws threatening the privacy of data. Moreover, cloud services can be distributed in more than one country, thus they can be under contradictory regulations because of the lack of international rules regulating the cloud environment.

- **High** level of risk
  - **R21. Subpoena and e-discovery**: when governments can confiscate physical hardware with shared tenancy, there is a higher risk of data disclosure for cloud customers.
  - **R22. Risk from changes of jurisdiction**: an unpredictable legal framework increases the exposure to subpoena law enforcement measures which can be in some case unacceptable.
  - **R23. Data protection risks**: the customer does not control the processing of the data, and he cannot know if the data is handled as he expects (in a lawful way for example).
- **Medium** level of risk
  - **R24. Licensing risks**: the customer must be sure that the intellectual property of his work is protected in the cloud providers environment.

Risks not specific to the cloud These are generic risks which are also targeting other services than cloud offers, however they are still relevant and can sometimes be even more important than in a non-cloud environment. Therefore they specifically need to be taken into account.

- **High** level of risk
  - **R26. Network management (ie, network congestion / mis-connection / non-optimal use)**: this could have an impact on the availability of the resources.
- **Medium** level of risk
  - **R25. Network breaks**: clouds are accessed through an internet connection, so this risk remains high and must still be considered.
  - **R27. Modifying network traffic**: this could affect the integrity of the transmitted data.
  - **R28. Privilege escalation**: when a user or an employee gets unexpectedly access rights through a misconfiguration or an intentional action of someone else.
  - **R29. Social engineering attacks (ie, impersonation)**, is an even more important risks with social networks where personal information can be easily obtained.
  - **R30. Loss or compromise of operational logs**, can be very important when log data is processed to determine access control policies.
  - **R31. Loss or compromise of security logs (manipulation of forensic investigation)**, in this case unauthorized access can no longer be proven.
  - **R32. Backups lost, stolen**, systems cannot be restored or data can fall into wrong hands.
  - **R33. Unauthorized access to premises**, including physical access to machines and other facilities.
  - **R34. Theft of computer equipment**, encryption keys or login information are often stored unprotected on computers.
  - **R35. Natural disasters**: redundancy of data centres and multiple network paths should considerably lower this risk compared to traditional infrastructures.

This list of risks is quite exhaustive in comparison to other listings such as in [CSA13] or in [CBT11]. But as stated previously, the values are static and have a generic value which can not be applied in any use case. Therefore, in the following we will explain why this cloud risk assessment needs some enhancements. In fact, we will even notice that some concepts from the information system security risk management domain are not completely compatible with cloud computing.

2.2.2.2 Cloud security risk management challenges and solutions

The main problem when performing security risk management processes in a cloud environment is that the whole information system is not always under full control, in opposition to a classical information system. This brings two major issues in play which makes a risk assessment and management difficult or sometimes even impossible.

Lack of control First, the cloud consumer has no longer control over the equipment hosting their assets and must integrate with an architecture defined by the cloud provider. He cannot implement the security controls that he considers as the most effective, or those proposed by his own risk assessment. Thus, there are some vulnerabilities a cloud consumer cannot foil and some risks that can only be accepted when outsourcing to the cloud.

In this sense, the first step to undertake before a cloud migration is to analyse if the company should or should not migrate to the cloud. Different assessment tools already exist for this kind of analysis as the Cloud Security Readiness Tool [21] published by Microsoft. It consists in a survey to help organizations review and understand their IT maturity level and their readiness to consider adopting cloud services. As already explained previously, an acceptable solution can be to avoid the cloud risks by not switching to cloud services.

Another solution is to change the **impact** of a security breach of a cloud service. In this vein, Jensen *et al.* [JSB+11] propose to split applications over multiple clouds through three different architecture patterns:

- **Replication of application**, to achieve redundancy in case of failure or compare different results of different services.
- **Partition of application system into tiers**, to separate the logic from the data.
- **Partition of application tiers into fragments**, none of the cloud services gives access to a full view of the processing logic or data.

Lack of information Second, the identification of vulnerabilities becomes a complex task for the same reason: the technical solutions are not under control of the cloud consumer and thus can even be unknown. Moreover, cloud providers may be tempted to conceal their vulnerabilities to not unnecessarily expose their services to attacks. Therefore, it becomes even difficult to assess the security risk when outsourcing to the cloud.

To overcome this problem different emerging standards are proposing metrics to assess the security of cloud services:

- The **Cloud Control Matrix** defined by the Cloud Security Alliance (CSA) [CSA14] proposes a list of 295 security controls to reduce security threats bound to cloud computing. The CSA advises cloud provider to publish those list which can give valuable information to potential cloud consumer about the security of those services. A platform is even available to download the lists of major cloud providers [22].
- The **Information Assurance Framework** published by the European Network and Information Security Agency (ENISA) [ENI09b] consists of a list of questions cloud providers should answer to give information about the security of their infrastructure.
- The **Common Assurance Maturity Model** [CAM10] is a project which goal is to provide a framework to potential cloud consumers to test if a cloud provider respects the consumers security requirements. It provides metrics over six key domains (governance, human resources, physical security, IT services, incident management and business continuity).
- The **Eurocloud Star Audit** [Eur12] is a service proposing to assess the security level of cloud providers. It provides an assignment of stars to cloud offers ranging from 1 to 5 stars based on a list of requirements cloud offers should implement.
- The **ISO 27017** [ISO15] is a code of practice providing a list of security controls cloud providers should implement in order to be certified by the standardisation organisation.

Basically, to assess the values of each risk, the **probability** and the **impact** of the risk must be previously defined. And those values depend heavily on the use case. On one side, the **probability**

21. `http://www.microsoft.com/trustedcloud`
22. `https://cloudsecurityalliance.org/star/`

depends of the selected cloud service. Indeed, each cloud provider makes different technical choices and thus implements different security controls. So, there are differences in terms of **vulnerabilities** and thus clouds more secure than others. On some clouds there are some risks which are more likely to happen than on others. On the other side, the **impact** depends on the type of deployed application. Indeed, each system is different and has different **objectives** in terms of security. Some companies will for example more focus on their response-time or their availability than others which will focus on privacy and the protection of personal information. Still, other may need a high quality of service with very accurate results. When using a cloud service for testing purposes, a theft of data will probably not be very relevant, whereas in other cases it is of crucial importance.

However, existing solutions are still emerging and thus their definitions are for the most still ongoing. Moreover, the solutions for both aspects (**impact** and **vulnerability**) are not formalized. This means that the measures against the **lack of control** cannot easily be automated. And concerning the **lack of information** this means that no general *metrics* can be provided (the models are mainly informal), which is an important aspect for quantifying the risk.

2.3 Business Process Management

Business process management is the domain that focuses on defining, managing and improving the processes of a company to deliver products adapted to its customer's needs. Business processes can be described in a formal way by using process models. Such models present the tasks and resources needed for performing the process and can have a graphical representation. Different standard languages exist for eliciting business processes through models. Business processes also have an implementation, which is the concrete way of performing the process. In modern and IT-based organizations, the implementation is supported by an information system on which the processes are deployed. When limited to software applications, processes can be automatically executed, with or without human interaction, using IT-services. Services may be locally or remotely accessible, for example through the internet, in this case they are called web-services. Therefore, in the context of computer science research, business processes are sometimes called web-service compositions, orchestrations (when centralized) or choreographies (when decentralized).

2.3.1 Definition

Usually, business process management refers to the traditional Business Process Life-cycle depicted in FIGURE 2.6, which is similar to the well-known Plan-Do-Check-Act approach. To support this life-cycle, business processes are represented using models. These models evolve through the life-cycle as each phase has its specific objectives. So there are different *abstraction levels* to represent a business process (sometimes also called *perspectives*).

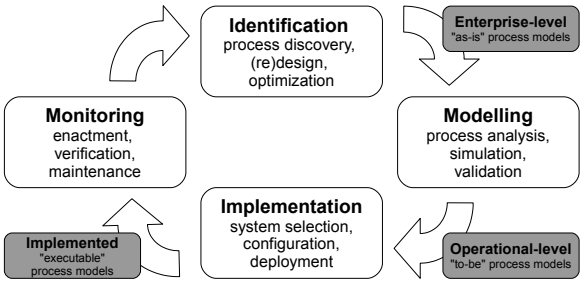


FIGURE 2.6 – Business Process Life-cycle drawn from [DRMR13], [Wes12]

2.3.1.1 The BPM life-cycle

As depicted in FIGURE 2.6, the BPM life-cycle usually consist in 4 phases. Some work may split the phases differently or simply name them otherwise, but they are globally in accordance to the life-cycle proposed below:

- **Identification** is the phase in which the organizational and technical environment is analysed to discover the business processes of the company. It also consists in challenging and reshaping the existing processes to optimize them. It is always the first phase of a new (or a first) cycle.
- **Modelling** is the phase where the processes can be graphically represented to detail the informal description defined in the previous phase. Some standard languages allow to simulate the model to detect some undesired execution sequences. Basically, this phase consists in validating the process in a theoretical way.

- **Implementation** is the phase which consists in transforming the abstract process model into an effective executable process. It can concern organizational changes for processes involving human activities. In case of IT-based processes it consists in the development and the configuration of software components.
- **Monitoring** is the phase in which the processes are executed. The monitoring of the execution is important for two main aspects. First, to check the compliance of the execution to the initially defined processes. Second, to gather data and evidences in order to start a new cycle for improving the processes.

2.3.1.2 The three levels of business processes

In the literature, there are often three levels of business processes, even if the content is often different depending on the authors [MAA11]. We decided to use the definitions coming from Ahmed *et al.* [AM13], which are quite similar to those given by Dreiling *et al.* [DRA05].

Definition 5 (Enterprise-level Business Processes) *Enterprise-level business processes are high-level processes which relate an organization to its business environment. It defines the business functions in a coarse-grained fashion. They typically specify the inputs and outputs of each process and their dependencies on other business processes.*

This abstraction level is mainly used to get an overview of the business processes of the company and their intra- and inter-organizational relations. Such type of processes are obtained after the *identification* phase and are used as input of the *modelling* phase. This view needs to be informal and has to hide as much complexity as possible.

Definition 6 (Operational Business Processes) *Operational business processes specify the activities and their relationships used to realize the business functions. The processes are modelled in a more fine-grained fashion, but disregarding any detail about their implementation.*

This level can be seen as "between" the abstract high-level and the detailed technical level. According to Dreiling *et al.* [DRA05], this perspective is intended for business analysts. Such type of processes are realized during the *modelling* phase and are used as input of the *implementation* phase. These processes can be defined for example in BPMN ([OMG11]).

Definition 7 (Implemented Business Processes) *Implemented business processes are the technical specifications to realize the activities of a business process. In an IT environment they are basically the software components supporting the execution of the process.*

This processes can be defined for example as executable BPMN or BPEL ([OMG11],[OAS07]). To obtain such processes, the operational processes are used as input of the *implementation* phase and transformed in different steps:

- First, the supporting systems are *selected* (infrastructure, platform, *etc.*).
- Secondly, if not already existent, the needed software components are *developed/implemented*.
- Then, the different systems components are *configured*.
- Finally, the processes are tested and then *deployed* in their production environments.

Currently, a growing market is those of tools supporting *Model-Driven Development*. Examples are Bonitasoft [23] or Webratio [24], which are development environments to automatically generate appli-

23. http://www.bonitasoft.com/
24. http://www.webratio.com/

cations from the business process models. Typically, they include a business process execution engine on which the process model is deployed and where the software components run. Other execution engines for running business processes also exist, examples are: Apache ODE [25], Activiti [26], BizTalk [27] or jBPM [28].

2.3.1.3 Standard modelling languages for business processes

As one of the main aspects of BPM is the modelling of processes, different standard languages have been defined to represent processes graphically. Here are the most important ones.

UML The Unified Modeling Language [OMG13] has been defined by the OMG and consists in different modelling layers (or perspectives). The *activity diagram* proposes a graphical representation of work-flows based on a semantic similar to *Petri-nets* and can thus be used for modelling organizational processes. However, UML activity diagrams are more intended for IS technicians rather than for business analysts or decision makers since the models are not really intuitive.

BPMN The Business Process Model and Notation [OMG11] is a graphical representation of business processes defined by the BPMI (Business Process Management Initiative). Based on flowcharts similar to *activity diagrams* from UML, BPMN also includes other concepts such as swim lanes, complex gateways or events. Since the version 2.0, models are meant to contain execution semantics in order to be executable. The strength of BPMN is that it provides an intuitive notation and thus provides a standard which is easily understandable for both, technical and business users. In this sense it greatly fulfils its main objective, to create *a bridge for the gap between business process design and process implementation* [OMG11].

BPEL Business Process Execution Language [OAS07] is an OASIS standard for defining and executing web-service compositions. In opposition to other languages, it does not include any graphical notation even if some vendors have proposed their own representation. The version 2.0 of BPMN also suggests a partial mapping between BPMN and BPEL. But BPEL is not meant to be human-readable and has only an XML specification. The originality of BPEL is that processes are represented as tree structures.

YAWL Yet Another Workflow Language [AAHR10] is a work-flow language developed by researchers from the Eindhoven University of Technology. It is also supported by an open source execution engine and graphical editor. Basically, YAWL extends the formalism of Petri nets to easily analyse processes. In opposition to BPEL, YAWL also supports human task allocation and is not limited to web service invocations.

Other modelling languages not listed here have a larger scope and are not limited to business processes. An example is the **ArchiMate** language which supports the description and the visualization of enterprise architectures. Thus, in addition to business processes, it also includes the modelling of organizational structures and technical infrastructures. However, such languages are out of our scope and globally reside on known aspects of the above presented standards.

25. `http://ode.apache.org/`
26. `http://www.activiti.org/`
27. `http://www.microsoft.com/en-us/server-cloud/products/biztalk/`
28. `http://www.jbpm.org/`

An interesting point is that all above mentioned languages are *task-centric*, which means that they focus on the activities performed during the process. But processes can also be *data-centric* to focus on the different states of the data objects during the process execution. Kumaran *et al.* [KLW08] presented an approach to transform a task-centric (or activity-centric) process into a data-centric (or information-centric) process to show the duality between the two models. Thus, in our work we consider those two type of notations as equivalent and that each representation gives a different perspective of the same model.

2.3.1.4 Reference architecture

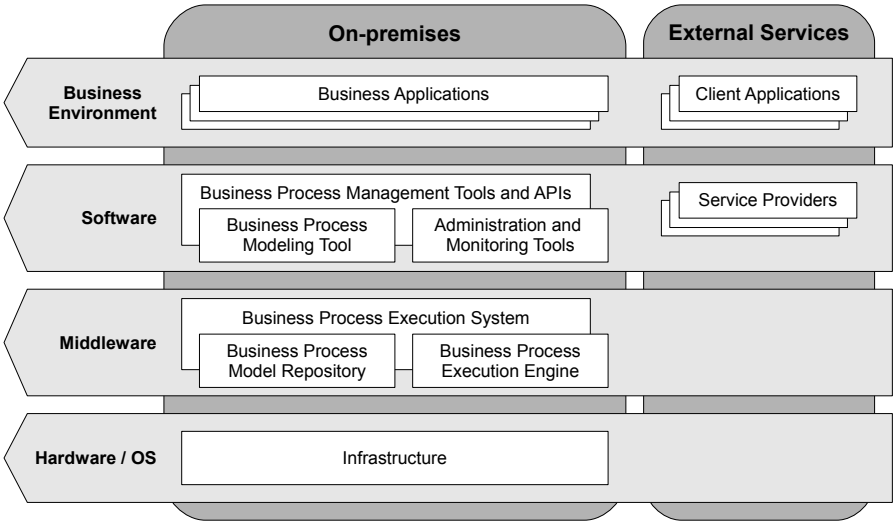


FIGURE 2.7 – Reference architecture of a Business Process Management System (from [DRMR13], [Wes12])

The reference architecture of an information system supporting business processes can be seen in FIGURE 2.7. It is drawn from [DRMR13] and [Wes12]. The architecture can be layered in four different levels presented in the following:

- **Business Environment**. Each process integrates within a broader environment providing its inputs or using its results. This can be the employees using and/or enacting the business processes, other applications or even external services. Since business processes are not standalone and independent, they fulfil a purpose defined in this business environment.
- **Software**. Are the tools used for supporting the definition, execution and monitoring of business processes. It can consist of *modelling tools*, *work-list handlers* and/or *execution log viewers*. Sometimes even APIs can be provided to give a programmatic access to interact with the processes. External service providers and their APIs are shown at this layer as they are needed to define the processes.
- **Middleware**. From a technical point of view it is the platform allowing the execution of the processes. Typically it consists of a *model repository* and an *execution engine*. Sometimes the modelling and monitoring tools are included directly in this platform resulting in a complete *Business Process Management System* solution.

- **Hardware / OS**. It is basically the infrastructure supporting the execution platform. It consists of the servers or virtual machines, storage systems and the network.

2.3.2 BPM, Cloud Computing and Security Risk Management

As already stated before, for IT-based business processes it becomes obvious to outsource them into a cloud environment for the same reason as any other information system: cost reduction and scalability. Moreover, business process models are an adapted tool for conducting risk assessments and managing security risks as they formally describe the processes of a company. Therefore, research efforts for applying security risk management methodologies on business processes for a cloud context seem to be natural.

2.3.2.1 BPM and Cloud computing

Different architectural propositions have been made for deploying business processes in cloud environments ([JLW+11], [FY10]). But the most detailed one has been made by the authors of [ALMS09], who investigate the different cloud computing delivery models and how they affect the outsourcing of a business process. Basically, they identify three deployment possibilities:

IaaS cloud providers only offer the infrastructure, thus the cloud consumer has to install a process execution engine to run its processes. Following the previously given reference architecture, only the *infrastructure* is controlled by the cloud provider. In this configuration, the consumer has the maximal flexibility concerning its configuration possibilities, while taking advantage of the cost reduction and the scalability of the cloud. However, he has to maintain the Business Process Execution System.

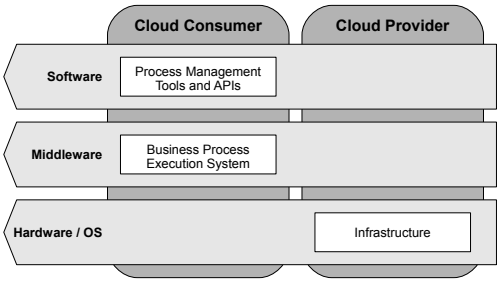


FIGURE 2.8 – IaaS architecture

The main advantage of such an **IaaS** architecture is that the user can even implement its own execution engine. In this sense, Muthusamy & Jacobsen [MJ10] propose an approach based on SLA requirements to describe and select different execution environments for their services. They are then able to re-deploy dynamically the different tasks of the process to efficiently use the available distributed resources. The approach also includes a cost model to optimize both financial and quality of service requirements. The execution engine can be considered as distributed, but controlled by the cloud consumer.

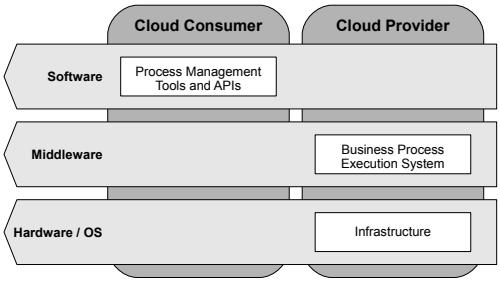


FIGURE 2.9 – PaaS architecture

PaaS cloud providers typically offer the execution system. In this configuration the cloud consumer runs directly its processes on the remote platform and provides only the business process models (*e.g.*, BPEL or BPMN). Monitoring tools or even modelling tools can be offered by the cloud provider, but the processes are under the full control of the cloud consumer (when to start/stop an instance, which service to use, *etc.*). The idea is that scalability issues due to the execution engine can be more efficiently managed by the provider.

Several solutions of such **PaaS** process execution engines are available. The most advanced platform is probably the one presented by Pathirage *et al.* [PPKW11]. In this paper they discuss the importance of multi-tenant architectures for executing business process in cloud environments to achieve the performance benefits of cloud computing. However, sharing the execution engine between multiple tenants creates security issues. They built a multi-tenant layer on top of Apache ODE to provide a business process execution engine able to run the work-flows of multiple tenants securely. Tenant's resources such as process models, instances, data and messages are isolated from each other. This platform is publicly available and the source code is open source [29].

A more academic system is presented in [MR14]. The authors implemented a modular and lightweight service-oriented work-flow engine supporting different languages (BPMN, BPEL, YAWL). The main advantage of this solution is that it can be directly embedded on any website. It supports modelling, execution and monitoring/debugging of processes.

The authors of [EJF+14] propose another original execution engine. They suggest to predict the needed resources of a business process based on its structure. In opposition to non-structured application, the execution sequence of a business process is known in advance. This can be valuable information to predict resource demand and scale the infrastructure accordingly. They built a BPM-aware resource controller which scales up or down the infrastructure based on the workload of preceding tasks and initially given scaling rules.

Focused on privacy aspects in process execution, the authors of [MF12] propose an architecture where process models are stored encrypted. Before being able to execute the processes, the execution engines has to require the decryption keys provided by the tenant of the process model. Even if the process can be revealed during the execution, it can be an interesting solution for multi-tenant cloud platforms.

SaaS cloud providers offer directly the processes. This is very interesting for re-using entire or parts of processes and offer them to other companies. The provider has a full control over the models, instances and underlying services to optimise their execution. The consumer can even consider the process as an external service (called BPaaS). Concerning financial costs, it is probably the most interesting configuration for the consumer, however he loses the flexibility of defining its process and must comply to the one defined by the provider.

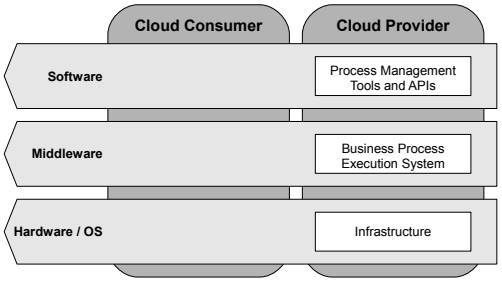


FIGURE 2.10 – SaaS architecture

The author of [Aal11] tackles the problem of **SaaS** business processes. He argues that the main benefit of cloud computing is the sharing of resources and software. However, sharing processes without the ability of customization is undesirable, since business processes are not natively reconfigurable once deployed. The proposition is to add an abstraction layer through *Causal nets*, to merge different variants into a single model. Moreover, it facilitates process discovery and conformance checking.

Focusing on the problem of privacy of processes provided through BPaaS, the author of [BBDA12] proposes a re-usability approach. By providing fragments instead of the whole process as a service, the process is not revealed in its entirety. Moreover, the consumer of such fragments has more flexibility for defining its processes.

29. `http://wso2.com/products/business-process-server/`

2.3.2.2 Security aspects in BPM

Lots of efforts have already been made to take into account security during the definition of business processes. Indeed, as early security is considered when designing an information system, as secure and cost-effective it will be.

Modelling security requirements in business processes Knorr & Röhrig [KR01] suggest already in 2001 that security requirements should be defined while modelling e-business processes. They propose to define levels (*Low*, *Medium*, *High*) over *security objectives* (*Confidentiality*, *Integrity*, *Availability* and *Accountability*) for each *phase* and each *party* involved in the process. They apply their framework on a sample shopping process and give examples of security mechanisms that could be used.

Jürjens [Jür02] proposes UMLsec, an extension to UML which includes security properties concerning *confidentiality*, *access control* or other security related constraints. These properties can then be checked at the model level by combining it with an *adversary model*.

The authors of [TBCB12] propose an extension to BPMN to model security constraints such as *confidentiality*, *non-repudiation* and *segregation of duties*. Thus, a process model can be annotated with security requirements. In the same vein, the authors of [RFMP07] propose a BPMN extension to add security attributes to existing elements. A *Message Flow* can be annotated with a *Non-repudiation* requirements, a *Data object* with an *Integrity* requirements, and so on.

The authors of [PGPM12] propose a framework to express security requirements on business processes by adopting multiple perspectives: the *social view*, the *resource view* and the *authorization view*. The security requirements are expressed through the usual security needs like *Integrity* or *Non-repudiation* and are then transformed into formal *Commitments* between actors. At this point, these cannot be transformed into requirements of a lower level language such as BPMN.

From security requirements to their implementation Wolter *et al.* [WMM08] add security goals to business processes through annotations, such as *Binding of duty* or *Signatures* for message flows. The interesting thing in this work is that the business process meta-model is extended with security constraints. The authors are then able to generate platform specific languages, such as XACML [30] policies or Axis2 [31] configuration files [WMS+09].

The authors of [TJG+11] also work at the modelling layer to define risk-aware business processes. They propose a formal model to describe the effects of *threats* and *safeguards* on the security attributes of the process activities (*Availability*, *Integrity*). This formal description is then used during a simulation phase to detect which safeguards are effective.

Another type of approach published in [MSSN04] considers that security requirements are already included in the business process model and that they can be derived from it. Thus, they propose a transformation script which derives an RBAC authorization policy from a BPEL process. However, other requirements not related to access control cannot be obtained with this approach.

Adapting business processes for security issues Another approach is to adapt directly the business processes according to the security issues threatening the process. In this sense, the authors of [AM14] propose to work with security risk-oriented patterns. They analyse possible attack scenarios on a business process and propose patterns to prevent them or mitigate their effect. What is interesting in this approach is that the BP model is used as the common communication tool between

30. https://www.oasis-open.org/committees/xacml/
31. http://axis.apache.org/axis2/java/core/index.html

the business analyst and a security analyst. They achieve a secure business process by exchanging attack scenarios and countermeasures in an incremental way.

In the same sense, but with different techniques, the author of [Fil12] proposes the idea of obfuscating business process models. By using different transformations, such as *Information hiding*, *Abstraction*, *Activity nesting* or *Attribute scrambling*, the final process can be shared without exposing confidential information. Even if it is still in its early phase, the global idea looks promising.

Secure execution of business processes in cloud environments As the cloud proposes different environments, offers can be divergent especially in terms of security. [CFBH07] presents a technique which resides in selecting the most adapted web service according to the specified security constraints. The approach defines a *Security Ontology* to map the security constraints defined in an internal XML language to the web services capabilities expressed through SAML [32].

More specific to the context of cloud computing, Watson [Wat12] proposes a multi-level security model to map work-flow activities to cloud offers. Each data element of the work-flow is associated to a security level, then the security levels of the activities are obtained through a Bell-LaPadula model [BL74]. A cost model is also included to select the most interesting deployment configuration in terms of cost.

The authors of [SLK09] propose an approach for selecting services during the process execution. The security of services are assessed through the Common Vulnerability Scoring System (CVSS) [33]. The *Protection goals* are specified through an *impact value* on the three CIA security attributes (*Confidentiality*, *Integrity* and *Availability*). In this approach, the costs of the selected services are also integrated to balance security risks versus financial costs.

Ouedraogo *et al.* [OBG13] propose a comprehensive model-driven approach to consider the security of business processes in a multi-cloud context. Based on a questionnaire that identifies the security requirements of a business process and its deployment architecture, the approach generates automatically security policies. These policies are enforced by a *context manager* which ensures that the classical security goals (*confidentiality, integrity, availability, non-repudiation*) are respected by invoking security services at run-time.

Other related work Some works consider other type of risks for business processes. For example, in [CLRA13] the authors take into account the risk of faults occurring during the execution due to the inputs given by the user. The risk value is calculated through the logs of past process executions. Interesting is that the approach helps to detect faults before their occurrence.

More generally, the authors of [MH06] consider risks threatening a correct definition of a business process. Thus, they analyse the BPM lifecycle to determine potential risks and propose different treatment strategies.

One approach presented in [WWHJ12] uses natural language processing methods to analyse the descriptions of the activities of a business process. By comparing them to previously defined *security patterns* the decision is taken which parts of the process can be outsourced or not.

We showed that risk-based approaches are meant to quantify security in information systems. In this sense, it is the first step to the automation of security management, since the security level of an information system becomes measurable. The formal representation of assets and their relationships

32. `https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security`
33. `http://nvd.nist.gov/cvss.cfm`

is brought by business processes and their management, another essential step for automating the risk assessment of a system. The last step necessary for the automation is the consideration of costs, since security decisions can only be made when balancing solutions against their implementation costs. Whereas costs are difficult to predict in on-premises and dedicated infrastructures (expenses are equivalent to investments), they become more easily measurable in cloud environments, where expenses are in the form of "pay-as-you-go" (operational expenses).

At first sight, the combination of these three domains should help to build systems with an automated consideration of security. The point is that there are no approaches or methods including all aspects of cloud computing, information security risk management and business process management at the same time. Existing approaches only cover one or two aspects of our research problem, but never its entirety. Either the cloud risks are not assessed over a formal representation of the system, and the assessment can therefore not be automated. Or business processes are used to manage the security of the system, but the approach is not adapted in a cloud context. Indeed, countermeasures cannot be implemented as wanted for cloud services, since they are not under full control. In the following, we present our contributions to this lack of solutions.

Outline of the contributions

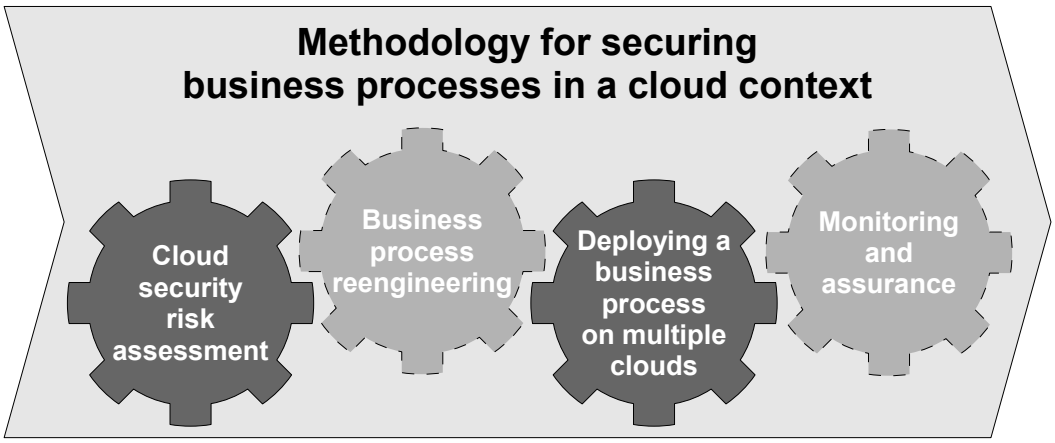


FIGURE 11 – Structure of the contributions of the thesis

The contribution section of this thesis consists in three parts, as illustrated in FIGURE 11. First, we introduce our global **methodology for securing business processes in a cloud context** (Chapter 3). It can be considered as an overview of our global contribution since it positions the following contributions in a wider context. Some methods, tool and techniques are introduced in this section that are not detailed in this thesis, but they are referenced accordingly (typically, process re-engineering like *obfuscation* or monitoring for *assurance* are only mentioned in our work). Two parts of this global methodology are more largely discussed in the thesis. First, the **cloud security risk assessment model** (Chapter 4) for evaluating the security of cloud providers. Second, how to fragment and **deploy business processes on multiple clouds** (Chapter 5) according to different cloud selection criteria.

Chapter 3

Domain alignment and methodological considerations

This chapter is mainly based on the contributions published in [GMG13] and [GMG14]. It can be considered as an overview of our global contribution. Since this thesis is at the intersection of three different domains (and scientific communities), an alignment of the different concepts of each field is necessary. Thus, we propose a common vocabulary and a generalized approach to manage security risks in a cloud context on business processes.

3.1 Distribution of the BPM life-cycle and levels

By clarifying the role of each actor during the BPM life-cycle (defined in Section 2.3.1.1), we can identify the possibilities and responsibilities of each cloud actor in the security risk management process and the type of information they exchange (as depicted in FIGURE 3.1). In the following we will detail first the role of the cloud provider, then the cloud consumer and finally end with the cloud broker.

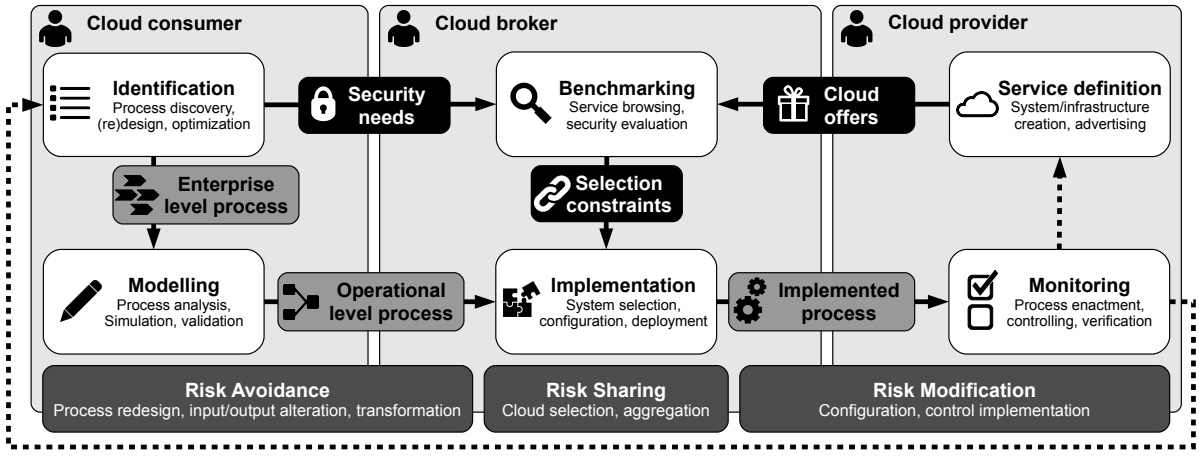


FIGURE 3.1 – BPM life-cycle distributed among the cloud actors and the associated risk treatment activities

Roughly speaking, **the cloud consumer** is responsible for **identifying** and designing his **enterprise level processes** (Definition 5 in Section 2.3.1.2). These are then transformed into **operational**

level processes (Definition 6) through **modelling** that can be transmitted together with the **security needs** to the cloud broker. The **cloud broker** uses these needs for **benchmarking** the different available **cloud offers** and generates **selection constraints**. The process models and the selection constraints are used by the broker to select the cloud services and finally deploy the processes (**implementation**). These **implemented processes** (Definition 7) can be **monitored** by the **cloud provider** to allow the cloud consumer to improve his processes.

This distribution of responsibilities allows to define the following coarse-grained alignments of risk treatment activities: the cloud consumer can **avoid** cloud risks by adapting his processes, the cloud broker can **share** cloud risks among different offers and the cloud provider can **modify/reduce** cloud risks by implementing security controls.

It is important to note that our contribution does not claim the requirement of an external and independent cloud broker positioned between a cloud consumer and a cloud provider (as presented in Section 2.1.1.3). This is not how our model should be seen, since the added value of the broker is not due to his impartiality. Quite the contrary: the broker can be impersonated by the cloud consumer. The objective here is to simplify the explanation of our approach by creating an abstract entity which separates the concepts only related to the cloud risk management process. This prevents us (and the reader) from mixing up the internal IS with the outsourced IS and avoids incorporating the internal security risks into the cloud risk management process. Once again, our goal is not to change the way the security risks of an internal IS, under full control of the company, can be managed. We focus only on the parts being outsourced to the cloud. Our approach does not prevent from applying a classic risk management method on the company's internal IS, as it does not prevent the provider from conducting a classic risk management process on its infrastructure. We only focus on the security risks generated by the creation of the cloud provider/cloud consumer relationship.

3.1.1 Cloud provider

In the context of deployable and executable business processes, the cloud provider supplies the services that support the business process' execution. So basically, they have access to the technical implementation of the process (or at least parts of it).

Typically, a cloud provider can change his system as he sees fit: he owns it (the infrastructure and/or the platform and/or the software). Therefore, he can implement countermeasures or security controls on his system to reduce security risks threatening his customers. This corresponds to the **risk modification** activity of the risk management process. By securing the provided infrastructure (or other types of services), the cloud provider reduces the security risks affecting the cloud consumer. But these measures are rather part of an offer than specific to a process. Indeed, most of the providers are not adapting their installations to comply to the requirements of one of their customers. They rather adapt their offers to comply to most of the requirements of their target market. Therefore, the security controls implemented in the provider's systems determine the security level characteristic to these **offers**. In this sense, the implemented controls become more a selection criterion of the provider, than a risk treatment strategy. FIGURE 3.2 illustrates this difference between a classic risk management process (1) and the one in a cloud context (2). Typically, a cloud consumer cannot modify the system provided by the cloud provider, in opposition to a classical architecture where he owns the system and can adapt it to his security objectives.

The provider influences technological choices to run a business process and has therefore a view of the **implemented** business process, still if it is not complete. However, even if it seems quite obvious, they cannot neither change the enterprise-level perspective of the business process, nor the operational.

Moreover, a cloud provider cannot avoid cloud risks because he is by definition exposed to them

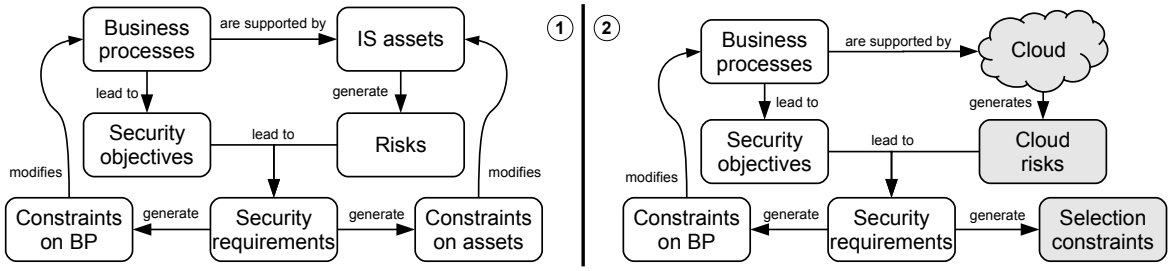


FIGURE 3.2 – Difference between (1) a classic RM process and (2) a cloud RM process

(due to the nature of the proposed services). Likewise, by subcontracting or outsourcing parts of his services (which corresponds to risk sharing), the cloud provider becomes a cloud consumer of another service ; this case is handled in Section 3.1.2.

One of the main actions of a cloud provider concerning security takes place during the **monitoring** phase of the BPM life-cycle, where he has to control his system and trace back to the consumer/broker each potential security breach. This means that, when considering the generic risk management methodology (see Section 2.2.1.2), a change in the provided services, an evolution of the system or even a modification in the terms of use, implies that the consumer/broker has to consider a new risk management cycle (or at least a re-assessment of the risks). This part of the global approach, which can be called **monitoring and assurance**, is not directly addressed in our work. Thus, in this thesis, there are neither a formalized approach nor security metrics which are defined to support the security of cloud business processes at "run-time". We focus our methodology on risk management at "design-time". However, even if it has not be done in the frame of this work, we consider that our approach could be extended to support such requirements. For example, the main reference that we use for defining security controls, the CSA STAR Registry [Clo14], already comprises the concept of *continuous monitoring-based certification* that could be integrated in our approach.

An additional point to notice is that cloud providers can generally propose better security contexts than in an on-premises infrastructure, since it is part of the core of their business activity. Thus, it is often interesting for consumers to switch to a cloud service for having a more secured system than they could define on their own.

3.1.2 Cloud consumer

The cloud consumer is obviously defining the **enterprise-level** processes since he designs his business strategies and the corresponding business functions. In a cloud environment, the idea is that a consumer can disregard any details about the implementation. In our case, even the selection of the execution system can be delegated to a cloud broker. Since we are studying the cloud risks threatening the consumer's processes, it is their responsibility to define the **security needs** of their processes.

As said in Section 3.1.1, cloud providers are by definition exposed to cloud risks, and these cannot be avoided. However, a cloud consumer can always make the decision to not outsource his processes, and thus to **avoid** the risks coming from the cloud context. Furthermore, the selection of one provider instead of another can also be seen as an avoidance strategy (the risks of selecting one provider are avoided by preferring another one).

Moreover, the cloud consumer can define some **constraints** emanating from his business strategy that have to be taken into account for the cloud service selection. Those constraints mainly stem from legal requirements. Some countries for example prohibit offshore locations for specific data types (in Luxembourg, data concerning the financial sector has to remain in Luxembourg). Regulations can

also impose particular certifications (like PSF in Luxembourg). Other requirements can even generate deployment constraints. When two tasks for example have to be enacted by two different persons (a separation of duty constraint), it can imply that those two tasks have to be deployed on two different cloud services (*separation* constraint). We propose the categorization of such constraints in three different types: **logical**, **organizational** and **informational** constraints.

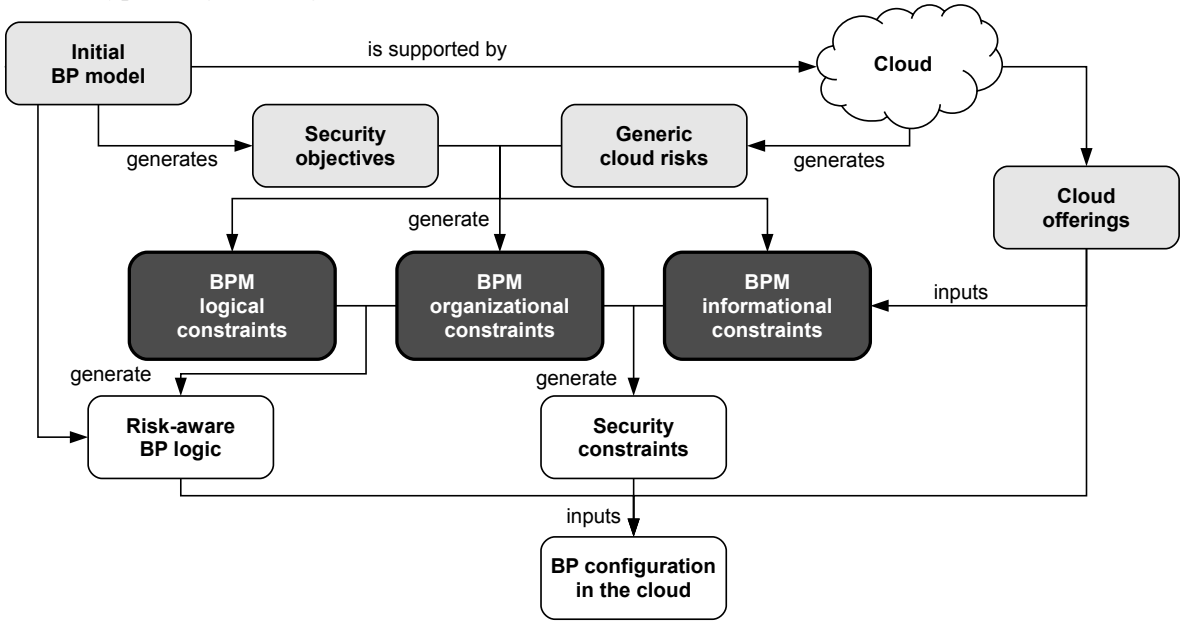


FIGURE 3.3 – Categorization of BPM security constraints for the cloud

Logical constraints They can also be called functional or behavioural constraints, as they impact the *control flow* of the process. Typically, the logic of the process is changed, as it behaves differently than it was initially designed: there are additional tasks and/or control flows. We consider following main principles:

- The **Split knowledge into several BP fragments** principle is especially useful for managing confidentiality at the BP level, typically for *know-how preservation* which is a highest level of requirement, if not the highest, for BP in the cloud context. The idea is to split the knowledge in several pieces in the objective of assigning resulting task/fragments to different clouds. In such a way, each cloud has only a partial view of the process. Only the root BP fragment which assumes the integration of contributing BP fragments has a global view of the process: it can be maintained on premises or assigned to a highly trusted cloud. This is achieved by splitting tasks/BP fragments in several tasks/BP fragments and adding the corresponding control flow.
- The **Separate logic and data** principle is especially useful for managing confidentiality at the logical level, typically for hiding some relationships between data and tasks which represent an important part of the *know-how to be preserved*. This principle is in some way yet supported by the separation of logical and informational levels in the BP modelling process, but the cloud context can request to enhance this property by splitting a task/BP fragment for introducing new tasks specific for data management.
- The **Group knowledge pieces into one BP fragment** principle is especially useful for enhancing *data integrity* by putting in the same place the more valuable assets so that it is easier to watch them. This is achieved by grouping several tasks/BP fragments in one task/BP fragment and adding the corresponding control flow.

- The **Replicate fragments** principle is especially useful to *verify* that clouds operate as promised (by comparing results and performance), and *that they deserve the trust put in them*. This is achieved by replicating task(s)/fragment(s) and adding task(s) to synchronize replicas. Replication can also be used to improve availability (through redundancy) but this is not central to our purpose.
- The **Add security management tasks** principle is to add non-functional tasks dedicated to security objectives in the cloud context. We do not think that these tasks are specific to the cloud and a taxonomy as this is defined in [GMR+12] can be reused. However, new needs, not anticipated in the initial BP model, can emerge due to the cloud context. For example, anonymization tasks can be defined to enhance *data confidentiality*, logs management tasks to support the verification of conformance of cloud executions, and others to compare the performance of replicas.

Organizational constraints In traditional BP settings, the BP organizational level defines for each task, the role (capacity) requested to execute the task. It also defines task assignment rules for constraining resource allocation (like separation/binding of duties), and it assigns tasks to organizational units (swim-lanes).

This remains in the cloud context, but the cloud itself can be constrained by organizational rules for achieving security objectives. Especially, these new rules provide security requirements for the cloud selection in the BP configuration process. They can be directly connected to requirements at the logical level: for example it seems a good practice to assign two BP fragments, required to be separated for preserving knowledge, to two different clouds.

We list here a representative, but not exhaustive, set of such rules:

- A **Separation of fragments** rule imposes two process fragments to execute in two different clouds. As its name indicates, its objective is to fragment knowledge and it is especially useful to support *Separation of knowledge* and *Separation of logic and data* decisions taken at the logical level. The difference here is that two BP fragments cannot be deployed on the same cloud offer, whereas the logical constraint does not enforce this “physical” separation.
- A **Co-location of fragments** rule imposes two process fragments to execute in the same cloud. As its name indicates, its objective is to group knowledge and it is especially useful to support a decision to group knowledge taken at the logical level. However, it is possible that two logically separated fragments can be physically located on the same cloud offer by combining the **split knowledge into several BP fragments** rule with the *co-location of fragments* rule.
- An **Impose retention of knowledge at premises** rule imposes a BP task/fragment to execute at premises, typically because it is a highly critical task/fragment.

Informational constraints Informational constraints are typically dependent on information coming from the cloud context (and thus the providers). Whereas logical and organizational constraints are context-independent, informational constraints will not modify the process model, but the directly the deployment configuration. At the informational level, a customer can constrain cloud selection in three main ways. She/he can:

- **Ban a given cloud** for executing a task/BP fragment because she/he does not trust it or it does not correspond to some regulations or internal principles (*e.g.* unacceptable location)
- **Impose a cloud** for executing a task/BP fragment, because she/he trusts it ; she/he has good experience with this cloud, or simply, it has a very good reputation. It is also possible that a

specific cloud offer is imposed through a regulation/certification, such as for governmental agencies.

- **Impose a level of security** a cloud must have for executing a BP task/fragment. For example, impose a minimum level of security a cloud must provide, either globally (for example, not less than 3 stars ranking in the *Eurocloud Star Audit* system for the cloud implementing BP fragment x), or regarding a specific risk (for example, grade 3 for *Governance* in the *Common Assurance Maturity Model*).

Of course, cloud consumers do not rely on an entire cloud-based information system. The security of their internal information system can and has still to be managed through a classic risk management approach. But this problem is out of our scope, as we consider that classic methodologies solve it. We only focus on business processes candidate for being outsourced.

In our context we consider that the broker is in charge of the implementation/deployment of the business process. Thus, the consumer has no technical view of the system and cannot implement countermeasures: implementation is completely disregarded. However, we argue that a cloud consumer can adapt his process in such a way that risks are limited. In short, cloud consumers can change the impact of a potential security breach by designing their processes otherwise. This is done with the help of the broker, since to do so, the risk has to be known from a prior risk assessment. More details about that are given in Section 3.2.

The phases during the BPM life-cycle where the main actions of the cloud consumer take place are **identification** and **modelling** which are respectively conducted by the company's managers and its business analysts.

3.1.3 Cloud broker

As the cloud broker's role is still emerging, it is rather unclear how far his expertise will go. According to the definition of Section 2.1.1.3, his main business activity will be the *system selection* step of the **implementation** phase.

Basically, the role of a cloud broker could be summarized as translating functional requirements (**operational-level processes**) and non-functional requirements (**security needs**) given by the consumer into cloud **selection constraints**. Consequently, the implemented processes, based on one or multiple cloud services, will be secure. A cloud broker can aggregate multiple services to fulfil the initial requirements, and distribute the risk among a set of different cloud providers. And as for the other requirements, the broker needs to ensure that the global security level of the distribution is acceptable. Therefore, the main risk treatment activity of the cloud broker corresponds to **risk sharing**. He is neither modifying, nor avoiding risks, but he tries to distribute them in such a way that risks are managed optimally.

Cloud brokers have neither the ability to change the business functions, nor to change the infrastructure, as they act as intermediation. But a very complete cloud broker could also provide *configuration* and *deployment* support while conducting risk **modification** activities. But these are limited to the layers above the used service (see Section 2.1.1.1):

- in the case of an **IaaS** architecture, the broker can counter the vulnerabilities coming from the *Business Process Execution System* and the *Process Management Tools and APIs*. It is up to the cloud provider to manage the vulnerabilities from the *Infrastructure*.
- in the case of a **PaaS** architecture, the broker can counter the vulnerabilities coming from the *Process Management Tools and APIs*. The cloud provider has to manage the vulnerabilities of the *Infrastructure* and the *Business Process Execution Engine*.

- in the case of a **SaaS** architecture, the broker cannot counter any vulnerabilities, only the provider can implement countermeasures to protect its *Infrastructure*, the *Business Process Execution Engine* and the *Process Management Tools and APIs*.

We even argue that a broker could advise a cloud consumer in some decisions as they are generally well documented on existing cloud services. In this sense a cloud broker could help the consumer to **avoid** some risks, as already suggested previously. However, the broker cannot make this decision on his own, he cannot change the business functions, this must be discussed and validated with the consumer.

In the next section we will focus on the cloud brokers role and the different actions he can take to securely deploy a consumer's process on one or multiple clouds.

3.2 Methodology for securing business processes in a cloud context

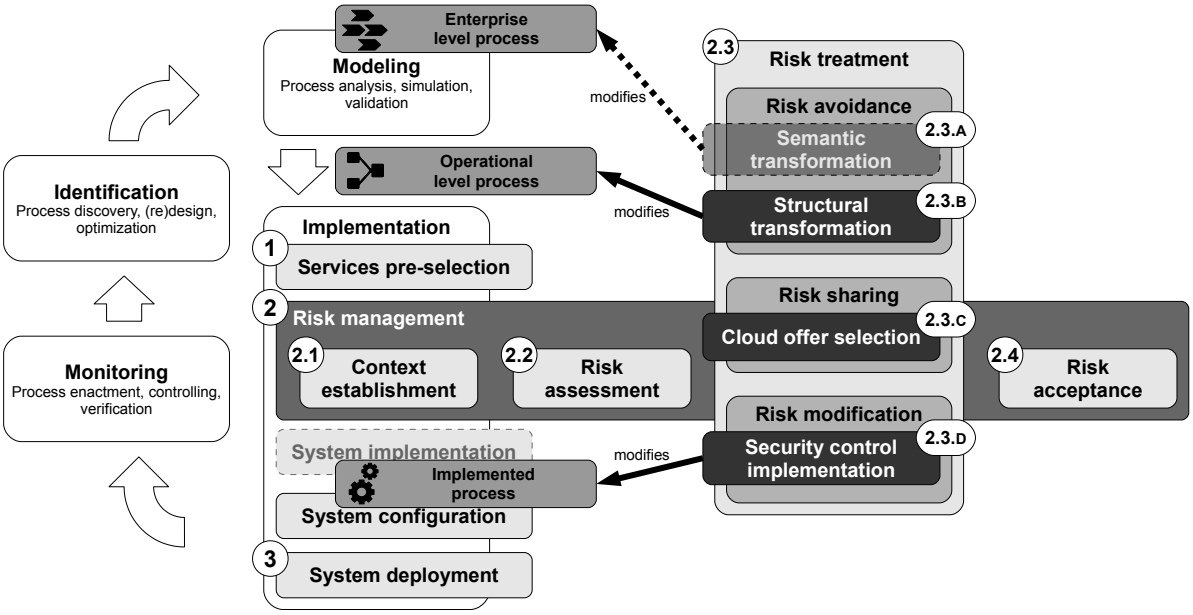


FIGURE 3.4 – Overview of our global contribution

As shown in FIGURE 3.4, we propose to align the BPM life-cycle with the three risk treatment strategies and the three previously defined cloud actors. These actors intervene in a cyclic way following the different phases of the BPM life-cycle. First a cloud consumer identifies and models its business processes at the enterprise-level. At this phase he is able to avoid some cloud security risks by "preparing" his processes for the cloud. Once modelled, the operational-level processes are transferred to the cloud broker who has to implement them. The cloud broker is distributing the risks among the different selected cloud providers executing the business processes. The broker can also somewhat avoid and reduce some of the risks through different actions which are detailed in the following. The cloud providers enact and monitor the business processes and have previously reduced some of the cloud risks by implementing countermeasures. The monitoring can help to improve the processes and their security by looping through the different phases once again (see FIGURE 3.4).

Our contribution is to adopt the perspective of a cloud broker and delimit the area on which he can help the cloud consumer to manage the security risks threatening his processes. We identify five

major stages: *services selection*, *risk assessment*, *risk treatment*, *risk acceptance* and *system deployment*.

3.2.1 Services pre-selection and context establishment

FIGURE 3.4-(1) The main role of the cloud broker is to help the cloud consumer to fulfil his *functional requirements* with adapted cloud offers. Thus, candidate cloud services are selected by the broker (which corresponds to the *system selection* step of the classical BPM life-cycle [DRMR13, Wes12]). These can come from a pre-established pool of services (generally, a broker has already negotiated with providers advantageous prices for his customers), or a dedicated benchmarking. Some specific security and legal constraints given by the consumer can already at this point lead to the exclusion of some services (such as geographic location, distrust or in-house expertise: typically the **informational constraints**, see Section 3.1.2). Once this set of pre-selected cloud offers has been defined, the risk management process can start as depicted in FIGURE 3.4-(2).

FIGURE 3.4-(2.1) The remaining pool of services forms the *context* of the risk assessment. This means that the security risk bound to the use of each of these services has to be evaluated. In accordance with the common risk management process, it is also at this point that it has to be decided how the risk will be assessed (based on which criteria the providers will be compared regarding security).

In opposition to a classic risk management approach, where the system is already known, here the risk assessment is even conducted on services which will not be included in the final configuration. Worse, the more services are assessed, the more probable it becomes to find the most secured configuration. This gives a strong motivation for our work, since the automation of the risk assessment represents an important saving of time.

3.2.2 Risk assessment

FIGURE 3.4-(2.2) By taking into account the **security needs** of the consumer and the selected services, the broker can perform the risk assessment. Each couple (asset, service) is associated with a risk value, which represents the security risk of deploying the given asset on the available service. This means that not all assets have to be located on the same cloud provider. The cloud consumer can deploy his system on multiple cloud services, to use them optimally in terms of capabilities, costs, security or other criteria. This fragmentation will depend on how much detailed the given security needs are:

- **company wide**, in which case using multiple services won't make a difference in terms of security. The security need is "global", different assets cannot be differentiated.
- **process wide**, in which case different processes can use different services to achieve a better security compliance. Some processes may need "more" security than others.
- **fragment wide**, a process can be split over multiple services, as some parts may have different security requirements. The process execution will be distributed.
- **task/data wide**, in which case the tasks of the process can be spread over different cloud services. The process can be completely broken down following cost or security parameters.

In Chapter 4 we present our cloud security risk assessment model which specifies in details how to perform such a risk assessment on business processes.

3.2.3 Risk treatment

For the three existing treatment strategies we defined four different transformations that the cloud broker can perform on the business process (see FIGURE 3.4-(2.3)). Depending on the type of transformation that is performed, they can modify either the **implemented processes**, the **operational level processes** or the **enterprise level processes**.

3.2.3.1 Semantic transformation

FIGURE 3.4-(2.3.A) One way to **avoid** some risks can be to change the semantic (logic, what the process is actually doing) of the process. Such type of transformation modifies the **enterprise level process**. This means that the global business function is altered in order to avoid one or multiple security risks emanating from the cloud. Of course the alteration cannot change the main strategy conducted by the company. Therefore this is not part of the core activity of a cloud broker, as he cannot alter on its own a business process in such a way.

But it can be interesting to advise the cloud consumer in some cases because a small change at the enterprise-level of a business process can dramatically change the global risk level of a business process, and at very little costs. A cloud broker can for example advise the consumer to use an in-house infrastructure rather than a public cloud. Only outsourcing a limited set of data can also be an interesting solution in terms of security. Another example can be a payment system: a broker could advise a cloud consumer to select another bank if his current one has a lesser secured payment system. But such changes cannot be decided by the broker alone, and must be decided jointly with the consumer since they directly impact the business.

3.2.3.2 Structural transformation

FIGURE 3.4-(2.3.B) Another way to **avoid** some risks can be to change the structure of the process. A structural transformation means that the process is changed, but without modifying the semantic of the process (the logic). Such type of transformation modifies the **operational level process**. The global business function remains unchanged, but the way this function is achieved is done in a different way. Such transformations are made at the operational-level of the BP. An example can be the splitting of some operations into multiple activities and the adding of separation of duties constraints. This is a task that a broker can typically do. He can also combine different services or add a specific layer in order to achieve the same goals but in a more secured fashion. Another example is *redundancy*: the broker can duplicate the process (or parts of it) on multiple clouds to increase the availability of the process.

Obviously, such transformations can also be a consequence of the **logical constraints** given by the cloud consumer (see Section 3.1.2).

An interesting perspective given by the fragmentation of processes is *obfuscation* of BPs. It is possible to decompose and deploy the structure of a process in such a way that it becomes difficult to discover what the process really does. By combining techniques like *separation of data and tasks*, *adding useless activities and message exchanges* or *replicating fragments for a random execution location*, the confidentiality of a process can be very well protected.

3.2.3.3 Cloud offer selection

FIGURE 3.4-(2.3.C) This is the core business activity of a cloud broker, called *service arbitrage*. Cloud offers have to be compared according to parameters as costs, security or quality of service. As some providers can offer services with a better security level than others, it can be interesting to

transfer the process to another location. As the security levels of cloud providers are often linked to the price of their services, it is important to balance the risk against the cost: a too secured provider could be too expensive, and on the contrary, the cheapest cloud would probably be not secured enough.

When handling one single process, the easiest way is to deploy the process on one selected provider. But, as explained previously, in some cases it can be interesting to partition the process into fragments and to deploy them onto separate cloud providers. As each activity of the process may not have the same security or functional requirements, it can be interesting to have a more heterogeneous deployment configuration in order to decrease costs or increase the quality of service.

Here, the broker has to take into account the **organizational constraints** given by the cloud consumer, as some tasks/fragments must be be co-located or separated.

A contribution to such deployment problems can be found in related work [FDGG14] and is detailed specifically for our use case in Chapter 5.

3.2.3.4 Security control implementation

FIGURE 3.4-(2.3.D) This corresponds to the mainly used risk treatment strategy: risk **modification**. Such type of transformation modifies the **implemented process**. The security risks threatening the business process can be sometimes easily reduced by changing or adapting the implementation of the process. If possible, the broker can integrate himself security controls into the system (encrypting the database for example). Other security layers (authentication for example) can be included to the system to increase the global security level. Otherwise it is still possible to reduce the risks by configuring the system correctly. An example could be to use SSL to secure all communication channels if this option is available for the selected cloud offer. This can obviously lead to an increase of the usage costs, which relates closely this option with the previous treatment possibility (*cloud offer selection*).

An interesting perspective currently emerging in the scientific community is the *homomorphic encryption* strategy. First published in [Gen09] by Craig Gentry, such approaches allow to process encrypted data by keeping it private. First promising solutions, such as CryptDB [PRZB12] or ZeroDB [34], propose database management systems where queries are run against encrypted data, without exposing any content of it. These systems can leverage considerably the security of cloud solutions, as they are almost immune against data breaches/thefts.

Those four different transformation are not given in any specific order, since the optimal actions to take can vary greatly for different use cases. However, as already said previously, the core activity of the broker consists in the *cloud offer selection*, thus it seems quite fair that most of the time it will be more interesting to first search for an appropriate service offer. Likewise, advising the cloud consumer to adapt their business process should be one of the last options.

3.2.4 Risk acceptance

FIGURE 3.4-(2.4) The last step is the acceptance of the risk, which can be reached after multiple loops in the risk management process. Each time an action is taken, the risk has to be re-assessed to determine if the risk can be accepted or not. This is usually done by defining a risk threshold: if all risk values are below this value, the processes can be securely deployed in their current configuration on the selected clouds.

34. http://blog.zerodb.io/a-peek-under-the-hood/

As implied previously, the last action a broker can take, when no acceptable solution can be found, is to advise the consumer not to deploy his processes to the cloud. Or, on the contrary, advise the consumer to revise his **security needs**, since they are too restraining for considering a cloud outsourcing.

3.2.5 Deployment

FIGURE 3.4-(3) Once the final deployment solution has been defined, the processes have to be deployed. Some brokers may include such services in their offerings. In case of a deployment of fragmented processes, the initial model needs first to be adapted to correspond to the selected configuration. The algorithm for an automated decomposition of processes can be seen in [FYG09].

Once deployed, the configuration should be monitored during the execution of the processes. This monitoring has three major objectives, which can be found in the BPM life-cycle (Section 2.3.1.1) and in the risk management methodology (Section 2.2.1.2):

- first, to be able to optimize, re-design, improve the current processes. The company needs to know what is working properly, what is not working as expected. This is an important step of the BPM methodology to improve the performance of the company. This is usually done by defining performance indicators.
- second, to verify that the cloud providers fulfil their contractual commitments, especially concerning the security of their offers (this is usually called *assurance*). The provider has to notify the cloud consumer of security incidents, services outages or possible malfunctions.
- third, to detect any change which could affect the deployed processes, either adversely (a decrease of a cloud's security level) or favourably (an increase of security, a decrease of costs). As the cloud context is dynamic, it is interesting to detect when it becomes more interesting to re-deploy the process in a more secure or cost-saving configuration.

3.3 Running example

In this section we come back to our running example introduced in Section 1.4. Typically, to outsource its processes to a cloud environment, the company would contact a cloud broker and request his help. Indeed, the company considers that it has not enough internal expertise concerning cloud computing for doing it on its own.

As defined in Section 3.1, the company (impersonating the **cloud consumer**) provides the business process model (FIGURE 1.3) to the broker. Typically, this model represents the **operational level** of the process. Additionally, the company provides the following requirements for the outsourcing in form of one **logical constraint** and one **informational constraint** (see Section 3.1.2):

The **persistence system (DB)** must be separated from the **Generate customer preferences** algorithm
The company considers that these two fragments are too important to expose them together, at the same location. The company believes that these are its two major assets, losing both of them (to a competitor or someone else) would be an existential threat.

The **payment_info** must remain on a cloud offer authorized for handling payment information
Due to legal requirements, the payment information of the customer has to be stored following a pre-defined security standard. The chosen cloud offer has to implement the PCI DSS certification [35].

The formalized **security needs** and the resulting security risk assessment is explained in detail in Chapter 4 and will not be introduced here. Moreover, the **cloud provider selection** step will be presented in Chapter 5 to show precisely how our approach can take into account security, costs and other parameters into account at the same time. However, we present some of the transformations on the BP models that the broker can perform to "prepare" the processes for the cloud:

- To reduce the risk of *insecure interfaces and APIs*, the broker proposes to use an external **Authentication or Registration** mechanism. Indeed, switching to a widely used identification service (such as Facebook, Google or OpenID) will considerably reduce the risks of a vulnerability in the login/registration interface. This transformation corresponds to a **semantic transformation** (*cf.* Section 3.2.3.1).
- To increase the availability and improve the scalability of the system, the broker proposes a *redundancy* approach. By replicating the **Add product to cart** task, it becomes less vulnerable to the risk of *resource exhaustion (under-provisioning)*. Moreover, if cleverly geographically deployed, it can improve the response-time: the used replica would depend on the proximity of the customer. This transformation corresponds to a **structural transformation** (*cf.* Section 3.2.3.2).
- Another interesting transformation could be to add an **Anonymization** task before archiving the order. Information such as the user's *address*, the products' *price* and the total *amount* could be removed. In such a way, confidential information would be efficiently hidden without affecting the overall process. This transformation also corresponds to a **structural transformation** (*cf.* Section 3.2.3.2).
- Finally, the broker proposes to split the persistence system in multiple fragments to counter the risk of *malicious insiders*. By storing the customers' personal information (*address*), their payment information (*payment_info*) and the order archive (*order*) at different locations, it becomes less probable that someone gets a full view of this data. Once again, this transformation corresponds to a **structural transformation** (*cf.* Section 3.2.3.2).

We can notice that some of these transformations/treatment will have consequences on other threats. As explained previously, the broker does more act on optimally distributing the risk (*risk sharing*) than reducing it. As an illustrating example with a probabilistic approach, we can take two providers A and B with a likelihood for the *data breach* threat of respectively 0.01 and 0.02. The chances of a *data loss* are respectively 0.002 and 0.001. Thus, the splitting of the database on those two providers will have following consequences:

- the probability that someone gains access to the full set of data (*data breach*) is equal to $0.01 \times 0.02 = 0.0002$
  Indeed, an attacker has to breach into provider A AND into provider B.
- the probability that some data is definitively lost (*data loss*) is equal to $1 - \big((1-0.002) \times (1-0.001)\big) = 0.00298$
  Indeed, it is the probability of losing some data either on provider A OR on provider B.

35. Payment Card Industry Data Security Standard: `https://www.pcisecuritystandards.org/minisite/en/pci-dss-v3-0.php`

Note that these calculations are correct under the assumption that the two providers are independent (the realization of a threat on one provider does not affect the other one). The final decision will depend on the importance that are given to the two threats (the *data loss* threat can for example be ignored). It also depends on the security needs defined for the affected data and the initially defined accepted risk threshold. In this example the broker decides to split the database, since overall the risk of a data breach is significantly decreased in comparison to the increase of the risk of losing data.

The resulting processes are shown in FIGURE 3.5. The **Authentication or Registration** tasks use an external *Identification Service* which also provides the *email* of the customer to the **Mailing process**. The database has been split in order to deploy it on distinct locations. The **Add product to cart** task has been replicated to deploy the two replicas on different cloud services (the details on the control flow have been masked for the sake of simplicity). The constraints given by the company are also annotated on the model with dashed boxes.

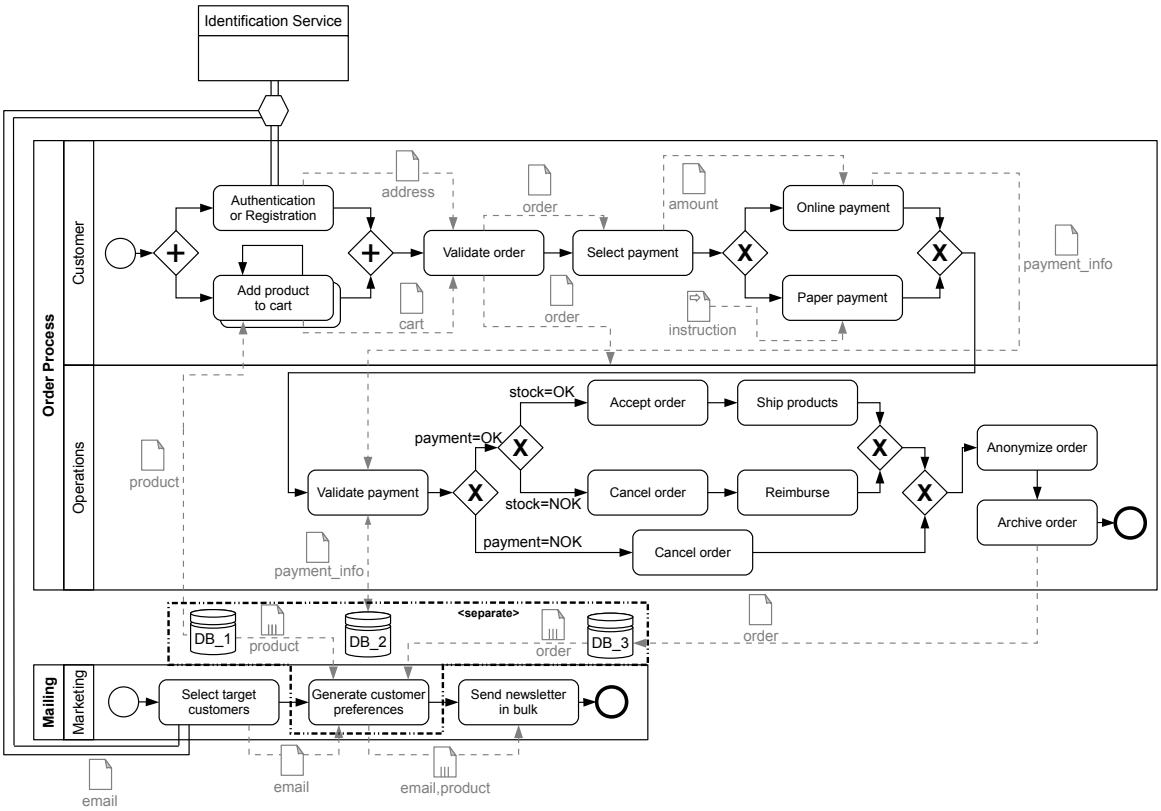


FIGURE 3.5 – Illustrating example - Modified cloud security risk-aware business processes

Once this configuration is obtained, either the risk is accepted as it is (in accordance with the company), or another transformation loop can be performed. For example, the broker could propose two additional transformations:

- use an *external payment system*, therefore the customer's payment information does no longer need to be stored. This would significantly decrease the risks affecting these processes. However, this is a transformation that only the customer can decide to do, as it will probably affect its business strategy (as an example, PayPal has in most of the cases higher fees than the rates that are negotiable with other banks). This transformation corresponds to a **semantic transformation** (*cf.* Section 3.2.3.1).

- the risk of exposing the entire order history could also be significantly reduced if not all orders were archived. As the archive seems only to be used for a statistical purpose, an idea could be to store only some orders (*e.g.* one of ten), which could be enough to process them statistically. Meanwhile, the sales revenue or the total amount of orders, which could be considered as sensitive information, would no longer be obtainable. However, this is also a transformation which impacts directly the business strategy and cannot be taken by the broker on its own. Once again, this transformation corresponds to a **semantic transformation** (*cf.* Section 3.2.3.1).

3.4 Conclusion

In this chapter we presented an approach integrating Business Process Management and Security Risk Management in a Cloud context. We defined the different actors, the process models they manipulate and the possible risk treatment strategies to secure a business process preceding a cloud deployment. The detailed methodology takes the perspective of a cloud broker, and categorizes the techniques he can use to lower cloud security risks threatening the business process. We illustrated our approach on the running example to describe more precisely what additional services a cloud broker can provide to cloud consumers.

In short, the points that need to be retained from this chapter are the following:

- the **cloud provider** executes the **implemented processes** and monitors them to inform the cloud consumer about their execution. Typically, he is responsible to **modify** the existing cloud security risks by implementing security countermeasures.
- the **cloud consumer** defines the **enterprise level** and the **operational level** processes. Typically, he can **avoid** cloud security risks by adapting his processes or by deciding to not migrating to the cloud. Additionally he can define three types of constraints on his processes:
  - **logical constraints** to enforce the control flow of the process.
  - **organizational constraints** to enforce the structure of the process.
  - **informational constraints** to enforce the type of selected cloud offer.
- the **cloud broker** can transform the **operational level** processes to adequately prepare them for existing cloud security risks. Typically, he **shares** the risk across different providers to distribute the risk as efficiently as possible. The transformations he can perform are the following:
  - **semantic transformations** can be advised to the cloud consumer to align its business strategies with the cloud risks.
  - **structural transformation** can help to efficiently distribute the risk across multiple cloud offers.
  - **cloud service selection** is its core business activity to deploy the process (or parts) on the most adapted offer(s).
  - **security control implementation** to reduce the risk of using an offer (typically through configuration).

Chapter 4

Cloud Security Risk Assessment

This chapter is mainly based on the contributions published in [GDG+14]. To perform the cloud offer selection based on the security risk of the possible deployment configurations, we have to define metrics. Here, we present a formal model to assess security risks of cloud providers and their offers. This model is then applied to business processes, however, we argue that it can be extended to support other types of representation and is not limited to BP models.

4.1 Model overview

Our approach relies on the different concepts presented in Section 2.2.2: **impact**, **vulnerability** and **threat**, which are usually used to define a security risk in information systems. Basically, our model relies on the assumption that cloud security risks cannot be assessed by neither the cloud consumer, nor the cloud provider on their own. The evaluation of the three concepts has to be split over the different cloud actors (see FIGURE 4.1).

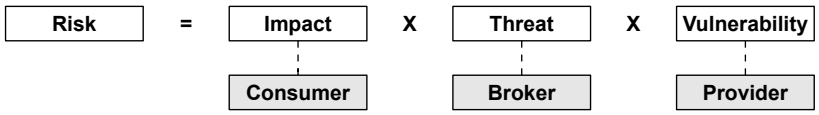


FIGURE 4.1 – Impact, Threat and Vulnerability are evaluated by respectively the consumer, the broker and the provider

Consequently we say that the impact can only be defined by the **cloud consumer**, since it is his assets which are affected by a potential security breach. When an incident occurs (like a *data breach* or a *denial of service attack*), the consequences directly affect the cloud consumer's assets. The cloud provider does not necessarily know if the consumer's data are for example confidential or if the service is only used for a testing purpose. Therefore, to properly assess the final risk value, the **impact** value has to be evaluated from the perspective of the cloud consumer.

On the other side, the vulnerabilities are given by the **cloud provider**, since it is his system which can have security flaws and allow an incident. The infrastructure, platform or software of the cloud service are under the responsibility of the provider. It is difficult for a cloud consumer to identify possible vulnerabilities, since mostly he doesn't even know the technology behind the used services. In opposition to a on-premises infrastructure, where the information system is under full control and can be investigated for security weaknesses by the company itself, in a cloud environment this has to be delegated to the provider.

The **cloud broker** can help the cloud consumer to define the impact value and the cloud provider to secure their offers, typically by defining the set of threats that have to be considered. Indeed, we consider that the expertise of the broker and his knowledge of the existing offers, gives him an non-negligible added value to cloud security risk assessment. Typically, when seeing the generic cloud risk assessment as published by the ENISA [ENI09a] or the CSA [CSA13], we consider that this information is independent from the two other values (the impact and the vulnerabilities).

Our model is more precisely illustrated in FIGURE 4.2. It defines the notions needed to evaluate the **impact**, the **vulnerability** and the **threat** before aggregating them into the final security risk value. The model can be divided into three sub-models, one **consumer model**, one **provider model** and one **broker model**. Once again, the cloud broker is not necessarily an external and independent entity, it can be considered as a role that the cloud consumer plays. This differentiation is made to facilitate the understanding of our approach.

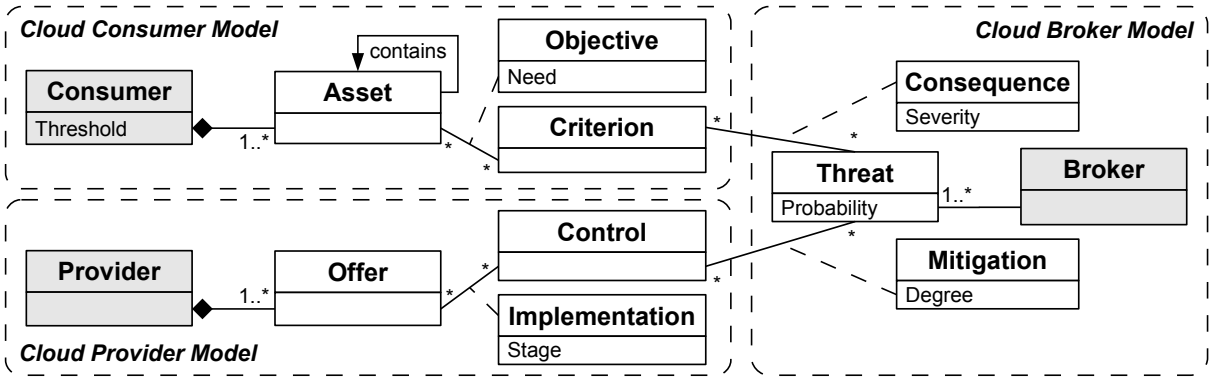


FIGURE 4.2 – Risk Assessment Model for multi-cloud environments

The cloud consumer has a set of **assets**, that are candidate for being outsourced on one or more cloud providers/offers. Assets can be of any type, software components, tools, models, or data elements. Assets should have a value, *i.e.* be of some importance for the company. The consumer defines a set of security **criteria** (typically *Confidentiality*, *Integrity* and *Availability*) on which he specifies his security **needs**. This association is called a security **objective**. Generally, the need corresponds to a value on a pre-defined scale to classify the assets in terms of their importance. It should be understood as follows: *an* **asset** *has the* **objective** *of fulfilling a security* **criterion** *which can be quantified by the* **need** *value*. This is based on the ISSRM domain model [May09] presented in Section 2.2.1.1. As an example, a security objective could be: "*passwords* (asset) should be kept secret (*high* need of *confidentiality*) within the company".

The cloud provider considers a set of **security controls** that he can **implement** on his infrastructure or services. Controls are safeguards or countermeasures used to prevent the occurrence of a security incident and increase the security of the installations. As an example, a firewall or an anti-virus software can be considered as security controls. Such controls can often be found in standards, certifications or best practices, especially in the case of cloud services. The implementation of such controls can be specified in terms of a **stage** that corresponds to a value on a pre-defined scale. In general, this information has a binary form (*is* or *is not*), but sometimes controls can be implemented gradually. In our model we differentiate the provider from its **offers**, since a provider can have multiple cloud offers, which do not necessarily have the same security characteristics. The relation should be understood as follows: *a cloud provider* **implements** *for each of his* **offers** *different security* **controls***, this relation can be quantified by the* **stage** *value*. This is based on common security risk

methodologies ([ISO11]), [AZ/04]) and existing cloud security approaches ([CSA13], [ISO15]).
As an example, "sensitive information are *encrypted* (control) with a AES-128 algorithm (*medium* stage of encryption) within the company", is a form of a security control's implementation.

In accordance with the cloud consumer, the cloud broker defines a set of cloud security **threats** to consider that could adversely harm the cloud consumer's assets. A threat is an event that is possible to occur and that would adversely affect the company.
On the one side, these threats have **consequences**, which can be defined through the security criteria (*e.g.*, some threats will affect the *confidentiality* of a resource, and others more the *availability*). This relation can be specified in terms of a *severity*, typically a value on a pre-defined scale. It should be understood as follows: *each* **threat** *has a* **consequence** *on the security* **criteria** *which can be quantified by the* **severity** *value*. This is also based on the ISSRM domain model [May09]. As an example, "a *denial of service attack* (threat) temporarily suspends (*high* severity in terms of *availability*) the provided service", is a quantified consequence of a threat.
On the other side, these threats can be **mitigated** by security controls which counter the security flaws permitting those threats. This relation can be specified in terms of a *degree*, typically a value on a pre-defined scale. It is a many-to-many relationship because one control can mitigate more than one threat, and similarly different controls can mitigate the same threat. It should be understood as follows: *each security* **control** *is intended to* **mitigate** *one or more* **threats***, this can be quantified by the* **degree** *value*. This is based on [CSA14] and [CSA13]. As an example, "systematic *background checks* (control) on new employees significantly reduce (*high* degree of mitigation) the risk of a *malicious insider* (threat) within a company", is a form of a quantified threat mitigation.

By combining these different information, we can evaluate the risk level for a given threat, when deploying a given asset on a specific cloud offer. The detailed way of doing it is formalized in the next section. But basically, the **objectives** and the **consequences** are aggregated to get a value for the **impact**, and the **implementations** and the **mitigations** are aggregated to assess the **vulnerabilities**.

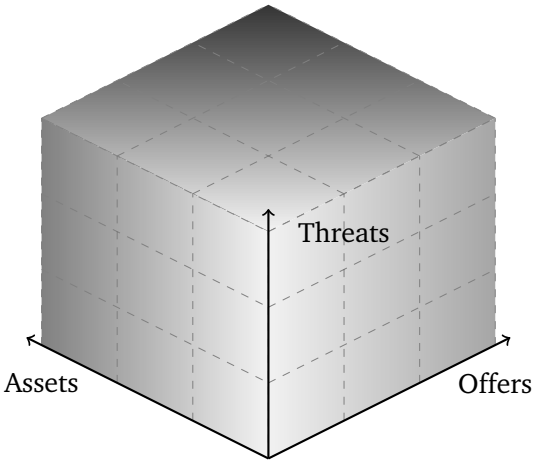


FIGURE 4.3 – Resulting 3D-grid of a cloud security risk assessment

The result of our cloud security risk assessment can be seen as a 3-dimensional grid where the dimensions are: the **assets**, the **offers** and the **threats** (FIGURE 4.3). Each cell of the grid has the associated risk value for the corresponding triplet. Thus, the risk of deploying a given asset on a specific offer is always detailed for all selected threats.
Once this grid is obtained, it is possible to merge the threats for simplification. Generally, only the maximum value is retained to get one single risk value, for one asset on one offer. In combination with the **threshold** it is then possible to exclude some deployment combinations (asset, offer).

4.2 Formal model

This section gives a detailed description of the concepts of our cloud risk assessment model and formalizes them. Two notions are introduced, the **coverage**, which represents the **vulnerabilities**

of the risk formula. The coverage consists in the formal aggregation of the *cloud provider model* of FIGURE 4.2. And the **harm**, which represents the **impact** of the risk formula. The harm consists in the formal aggregation of the *cloud consumer model* of FIGURE 4.2. The **risk** is defined by combining these two notions with the **threat's probability**.

4.2.1 Cloud provider: implementing security controls on offers

First, we formalize the concepts related to the cloud provider in Definition 8.

Definition 8 (Cloud Provider) *An entity which can hold multiple cloud offers. It is responsible for the implementation of security controls, to protect its offers from attacks, to prevent incidents and to comply with regulations (e.g., controls defined by the CSA [CSA14] or the ISO 27017 [ISO15]). It is specified as follows:*

Offer*, is a set of cloud offers available to subscription on the provider.*

Control*, is a set of security controls generally given in standards or certifications to prevent security incidents.*

Implementation *: Offer × Control → Stage, defines how a security control is implemented by an offer. The Stage can give a more fine-grained description than only a binary is/is not.*

Instead of focusing on vulnerabilities, as in [SLK09], our risk assessment approach focuses on security **controls**. We argue that cloud providers may be tempted to conceal the vulnerabilities of their offers, to not unnecessarily expose their services to external attacks. Publishing security flaws can be of great interest in the case of open source software, but may be very counter-productive for a cloud provider. It would become easier for attackers to target this provider. Therefore, we focus on security controls, which are more likely to be published. An example is the Cloud Security Alliance (CSA) which defines 295 controls in its Cloud Control Matrix [CSA14]. Providers can publish the controls they implement on a publicly available platform: the Security, Trust and Assurance Registry [Clo14]. Due to a lack of space, we do not show an exhaustive list of these controls, but examples are:

- *AIS-01.2* Do you use an automated source code analysis tool to detect security defects in code prior to production ?
- *BCR-01.1* Do you provide tenants with geographically resilient hosting options ?
- *EKM-04.3* Do you store encryption keys in the cloud ?
- *IAM-07.5* Do you provide the tenant the ability to declare a disaster ?
- *IVS-01.4* Are audit logs centrally stored and retained ?
- *TVM-03.2* Is all unauthorized mobile code prevented from executing ?

At the time of writing, these controls are grouped into sixteen different categories such as *Application & Interface Security* (AIS), *Data Centre Security* (DCS), *Identity & Access Management* (IAM) or *Infrastructure & Virtualization Security* (IVS). Due to the dynamism of cloud computing and the daily discovery of new security breaches/attacks, such lists evolve frequently.

In our approach, we indicate "how" such a control is **implemented** on a provider's offer with a **stage**. Basically, if the offer implements the considered security control, this value is equal to 1. If it does not implement the control it is equal to 0. This choice is recommended by the CSA in [CSA13], which considers that this information should be given in a binary form. However, in our approach we prefer an interval from 0 to 1 for more precision. Indeed, a control can be "partially" or "meant

to be" implemented, which can constitute a more complete information than a simple is or is not. We observed this while examining some of the published self-assessments on the STAR Registry: many providers precisely describe (in a textual form) how the control in question is implemented. Sometimes, only a subset could be considered as implemented. Moreover, some controls can never be considered as "completely" implemented, as it is quite impossible in practice (*e.g.*, the above mentioned *TVM-03.2* control can never be guaranteed, as new technical vulnerabilities allowing this can appear at any time).

An interesting point here is that it is "easy" for cloud providers to hand over these information to a cloud consumer. It is part of their job to implement countermeasures and prevent security incidents. Often, their security management is already driven by standard guidelines, so there is no additional work to do. Filling out such forms should be a routine task. Moreover, a provider does not supply any direct information about how an attacker should behave to exploit a vulnerability. In this sense, vulnerabilities can only be assumed or guessed indirectly. This way, providers are more open to transparency than when it comes to publish directly their vulnerabilities.

4.2.2 Cloud consumer: defining security objectives on assets

The concepts related to the cloud consumer are formalized in Definition 9.

Definition 9 (Cloud Consumer) *An entity which browses the offers from cloud providers (or cloud brokers) and subscribes to one or more services adapted to its functional and non-functional requirements. It is specified as follows:*

Asset, *the set of business assets of the consumer.*

Criterion, *a set of security criteria, typically {Confidentiality, Integrity, Availability} (sometimes Authenticity and Non-repudiation can be added).*

Objective *: Asset* $\times$ *Criterion* $\rightarrow$ *Need, defines security needs. It gives a description of the "quantity" of security needed for a given Criterion by an asset.*

Threshold *:* $\Omega \rightarrow$ *Level, defines a global acceptable risk level.* Ω *being the whole system, meaning that this threshold is defined globally.*

A security **objective** helps to determine the impact of a security risk. It is frequently defined in terms of the following security **criterion** (CIANA) [NIS02]:

- *Confidentiality*, the access to the data is restricted to those who are authorized to and cannot be read by another entity,
- *Integrity*, the data must remain consistent during its entire life-cycle and cannot be modified by an unauthorized entity,
- *Availability*, the data must be accessible when it has to be used, this is important to guarantee the proper working of an information system,
- *Non-repudiation*, the parties involved in a transaction cannot deny being involved in it, this is important for tracing responsibilities in case of an incident,
- *Authenticity*, the data (but also the communication channels) must be genuine, it must be guaranteed that the accuracy of the information can be trusted.

However, our model is not constrained to these references, one could also work with for example STRIDE [36] or other criteria defined for a specific use case. Indeed, there are many different reference models for information security (the Parkerian Hexad [37], IAS-octave [38] or the 33-NIST principles [39]). In this thesis we selected the CIANA criteria due to its popularity. We do not enter in a comparison of those references to find the best, we rather argue that they are tribute to subjective preferences and often depend on the use case on which they are applied. Some business domains even define their own security criteria to put the focus on their specific needs more easily. In some other use cases, the five CIANA criteria are limited to three: Confidentiality, Integrity and Availability. In this sense, all models, formulas and examples of this thesis can be adapted to use another reference model for defining security criteria.

These criteria must be fulfilled to ensure the security of a technical solution and its data [NIS02]. In existing risk assessment methods, different levels of security **needs** are usually expressed on business assets to constitute a security **objective**. Basically, some assets may have a "high" need of *Confidentiality* or a "low" need of *Availability*. Whereas other ones have a very "low" need of *Confidentiality* and a "high" need of *Availability*. But not all assets have the same requirements, since this would either generate too severe security constraints or at the contrary not severe enough. Generally, these variable needs of security can be expressed through different scales, rarely normalized. In our approach we define these values on a scale of [0, 1], with 1 being the highest possible value and 0 no need at all. Thus, an example for *Confidentiality* could be the following scale : $\{Public = 0, Restricted = 0.5, Secret = 1\}$.

The **threshold** is a global value used for the final cloud provider selection. It indicates the acceptable level of risk, below which the risk will be retained. It determines which cloud provider can be used and which not.

In our approach, where we work on business process models, we annotate the processes with these security needs in order to later decide where to deploy which fragment of the process. One question is, where to put these annotations, or what has to be annotated on process models. So typically, how to identify the assets.

4.2.2.1 From data-centric to task-centric security objectives

One key element, already pointed out in Section 2.3.1.3, is that business processes are mainly **task-centric**. This means that most of the modelling languages focus on the process itself (*i.e.* the tasks and their execution order) rather than on what is produced (*i.e.* the data and its states). Data-centric business process models exists (a survey of so-called artifact-centric models is given in [Hul08]), and the authors of [KLW08] present an algorithm to transform task-centric processes into data-centric processes. But in the context of cloud computing, where offers are described in terms of services (*cf.* IaaS, PaaS, SaaS), it is easier to perform a process outsourcing based on the tasks than on its data. Therefore, our approach selects cloud offers and deploys processes based on their tasks (typically a task represents a web-service call).

However, security needs are usually expressed on data objects ([GBO+07], [ZSM+10]). Indeed, information security is, as its name implies it, focused on information. This is why, underlying models for managing, assessing or guaranteeing security are always **data-centric**. Security risk management methodologies are aligned with this vision, since they work on *assets*. It is difficult to consider a task,

36. Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, Elevation of Privilege (a system developed by Microsoft [HLOS06])

37. http://en.wikipedia.org/wiki/Parkerian_Hexad

38. http://en.wikipedia.org/wiki/Reference_Model_of_Information_Assurance_and_Security

39. http://csrc.nist.gov/publications/nistpubs/800-27A/SP800-27-RevA.pdf

a process or a service as an asset, because they are often not tangible. Thus, it becomes easier to annotate the data of an information system with security objectives. As a matter of fact, security criteria such as *Confidentiality* or *Integrity* are appropriate for annotating data elements of a business process, but are difficult to interpret on tasks. So in our approach, we express security objectives on the data objects of the process model.

However, in processes the data objects can have different states: objects are evolving during the execution of the process. Data-centric processes express this fact very well in the sense that the "nodes" of a task-centric process model are often the state of the data element. As an example, during a buying process, the *order* has a *status* that evolves: it can change from *awaiting* to *paid* or *shipped*. Moreover, an object can contain information that are existing at some point of the process, but do not exist at other points. Thus, the security need of an object can also evolve during its life-cycle. Typically, the *confidentiality* or the *availability* of a data object has not to be always the same at any point of the process. Kumaran *et al.* [KLW08] identify these states based on where the data is used during the process execution: each time a data object is output of a task, it gets a new state. In this way, all evolutions of the data object are tracked. So, in our approach we identify as assets the states of the data objects. This corresponds to the data-flows (or associations) of a process modelled in BPMN.

However, we need to relate data-centric security needs to the tasks of the business processes (since tasks are deployed on clouds, not the data objects directly). This is achieved by applying a simple access control model based on the Bell-LaPadula model presented in [Wat12]. Basically, the security need of a task is defined as the highest need of the data objects it handles. This is formalized in Definition 10.

Definition 10 (Task-centric security need) *For a task t_i, in- and out-going data objects d_j of t_i and the security criterion o_i, the task-centric security need is defined as follows:*

$$Objective(t_i, o_i) = \max_{d_j \in data(t_i)} \big(Objective(d_j, o_i)\big) \tag{4.1}$$

This formula is motivated by our risk-oriented approach. Since we use these needs to evaluate the **impact** of the risk formula, it makes sense to speak of security needs for a task. It corresponds to the impact that a security incident would have on the data of this task. In this sense, our task-centric security needs must be understood as the security needs for the data processed by this task (and not of the task itself). We argue that the data objects processed by a task, will be present at some point on the cloud offer that executes this tasks. And the impact that a security incident on this offer can have is defined by the data that is present on this cloud offer.

Therefore, the needed security on this offer, is at least as high as the data that has the highest need handled on this offer. For example, if a cloud service handles two data types, one that need a *high* level of confidentiality, and one that has a very *low* level of confidentiality, then the service will have a *high* need of confidentiality.

In our approach we do not distinguish between in- and out-going data objects of a task. We argue that in both cases, the data object will be present on the cloud offer, and thus the translation of the need is the same. Some approaches, like that presented in [Wat12], apply a *no read-up* and *no-write down* rule (so it differentiates between in- and out-going data elements). But such an approach, where the security need corresponds more to a "clearance level", is more adapted for a classic access-control model. In the cloud context, risks that apply to in-going data also apply to out-going data.

4.2.3 Cloud broker: threats, mitigations and consequences

The concepts related to the cloud broker are formalized in Definition 11.

Definition 11 (Cloud Broker) *An entity which can provide three types of services [LTM+11]: enhance existing services like security (service intermediation), combine multiple services (aggregation) or measure different providers and select the best (arbitrage). It is specified as follows:*

Threat, *is a set of cloud security threats. This set should be generic for all use cases, since cloud threats are by definition present when using cloud services.*

Control, *the same set of controls than those considered by the cloud provider.*

Criterion, *the same set of criteria than those chosen by the cloud consumer.*

Consequence: *Threat* $\times$ *Criterion* $\rightarrow$ *Severity, indicates how the threat has a consequence/affects the criterion.*

Mitigation : *Threat* $\times$ *Control* $\rightarrow$ *Degree, indicates how a control mitigates/reduces the threat.*

As explained in Section 2.2.2, there are different cloud-specific security **threats** (*e.g.*, *data breaches*, *data losses*, *malicious insiders*, *etc.*) that can create at any time a security incident. To calculate the risk of these threats they need to be related to the consumer's assets, to evaluate their "importance" (*i.e.* the **impact** of the risk formula). In our approach we argue that each of these threats have a **consequence** on one or more **security criteria**. This means that the occurrence of this threat would adversely affect the criteria, and thus negate the security objective. When using for example the CIANA list of criteria and the CSA list of threats, each threat can be related to one or more criteria (see [CSA13]):

- *Denial of service attacks* have a meaningful consequence on *availability*.
- *Data breaches* have only a consequence on *confidentiality*.
- *Data losses* however have a consequence on *availability* and *non-repudiation*.
- *Malicious insiders* can have an important consequence on all five criteria.

In this example the consequences are binary relations which can be represented in a matrix relating the threats to the criteria. However, in our approach we consider that the consequence can be defined in a more fine-grained fashion with a $[0, 1]$ interval. Indeed, some threats can have a more important effect on one criterion than on another. *E.g.*, the *data loss* threat is more likely to impact the *availability* of a resource than its *non-repudiation*, even if it is not impossible. An example of such a consequence scale could be the following: $\{Negligible = 0, Related = 0.5, Significant = 1\}$. Once again, this can be interesting in some use cases, where such tweaking can improve the final results. Another argument is that often, most of the recorded threats affect all considered security criteria, thus it becomes interesting to weight the consequence for each criterion.

Typically, this mapping of security criteria to cloud threats are made by a community of security experts (as it has been made for the CSA [CSA13]). Similar reports exist (such as the ENISA cloud security report [ENI09a]) from which a cloud broker can build this knowledge base. Since such type of mapping can be considerable and have to be continuously maintained, we consider it as a substantial added value for the broker's role. Indeed, we noticed that the list of threats is constantly evolving to reflect the prevailing cloud security issues and cannot be considered as static.

The other value that has to be taken into account for assessing the risk is the vulnerability. Basically, a cloud provider can be considered more secure than another one if he implements more security **controls**. Thus, it is interesting for a cloud provider to implement many countermeasures to reduce the probability of security incidents as much as possible. However, this is not systematically true, because security controls are intended for a specific purpose, *i.e.* reduce a given threat. This is why we define the concept of **mitigation**: threats can be mitigated by one or multiple security controls. As an example, the CSA relates security controls to threats in [CSA13]:

- *Data breaches* are mitigated by the controls *EKM-02.3 (Do you maintain key management procedures ?)* and *IAM-12.7 (Do you allow tenants to use third-party identity assurance services ?)*.
- Control *EKM-02.3* also mitigates the *malicious insider* threat.
- Whereas control *IAM-12.7* additionally mitigates *account or service traffic hijacking* and *shared technology vulnerabilities*.
- The control *BCR-08.1 (Are security mechanisms and redundancies implemented to protect equipment from utility service outages ?)* is essential for mitigating *denial of services*.

Similar to the consequences, the CSA defines binary relations for **mitigations** while we consider that this information can be improved. We argue that some controls are more effective than others for mitigating a threat. A very simple example, *redundancy* and *weekly backups* mitigate the same threat: *data loss*. However, the first one is way better than the second. Thus, one "good" control can easily be equivalent to 5 or 6 "moderate" controls. In this sense, we define a $[0, 1]$ interval analogous to that of the consequences: a value of 0 meaning that the threat is not mitigated at all, a value of 1 meaning that the threat is completely mitigated.
Optimally, all control's mitigation values for a given threat should add up to the value of 1, meaning that when implementing all these controls, the threat has no longer a chance of happening. In practice, this is quite unrealistic, since there are no existing configurations that completely disable a threat. There is, and there will always be, a possibility to make a threat happen. Therefore, a value of 1 should be understood as the "best known configuration to have the threat's probability as low as possible".
One workaround could be to define values for the mitigations in such a way that the value of 1 is never attainable. However, this would not only be a limitation to our approach, but would also create a subjectivity issue: what does a maximum mitigation value of 0.9 mean in comparison to one of 0.8 ? That there are fewer existing security controls to counter it, and thus the threat is globally more probable ? This information is already contained in the **threat's probability** which will be presented in Definition 14.
In addition, we argue that this approach allows us to have another benefit, because in practice, some controls may be ineffective when another one is already implemented. Indeed, there can be an overlapping of some mitigations, meaning that their combined effect adds up to more than 1. It reflects the fact that at some point it becomes useless to implement more security controls than necessary. An example, encrypting data on transit, while all communication channels are already securely encrypted does not make lots of sense. It won't make any harm (even if in this case it will slightly increase the global response time), but it does not add "more" security. Since formally, it does not make sense to have an aggregated mitigation of 1.1 of 1, the mitigation is maximized at 1 (as said previously, at some point, some controls become useless).

In order to quantify the risk, we combine the previously defined concepts (**implementation**, **mitigation**, **objective** and **consequence**) for calculating two different values for each threat: the **coverage** of a provider, and the **harm** on an asset, as defined below.

4.2.4 Coverage: control implementation and threat mitigation

In this section we define the concept of **coverage** (Definition 12).

Definition 12 (Coverage) *A score calculated for a given provider and a given threat. It is calculated with the security controls implemented by the provider and their mitigations on the threat. Formally,*

$$\begin{aligned} \textbf{Coverage}: & Offer \times Threat \rightarrow Score \\ & o, t \mapsto Coverage(o,t) \\ & = \min\left(1, \sum_{c \in Control} \Big(Implementation(o,c) \times Mitigation(t,c)\Big)\right) \end{aligned} \tag{4.2}$$

Mostly, a provider who implements many controls will be more secure than one who implements fewer controls. In our approach this score allows us to compare the response of a provider to a specific threat, it corresponds to a percentage of implemented controls that are needed to mitigate that threat. In [Clo14] for example, the CSA shows for each provider, their implemented controls. So, if the CSA gave 10 controls to mitigate a threat, and a provider implemented 5 of them, he gets basically a coverage of 0.5 (50%).
In our approach, this value can be influenced by the **mitigation** and the **implementation**. As said previously, some controls may be more effective for mitigating a threat than others. And a control can be "partially" implemented and thus be not fully effective. For example, given that controls *EKM-02.3* and *IAM-12.7* mitigate *data breaches* with a value of respectively 0.4 and 0.2, and that a provider implements them respectively *completely (= 1)* and *partially (= 0.5)*, the coverage of this threat t_i for this provider p_i would be:
$covg(t_i, p_i) = \min(1, 1 \times 0.4 + 0.5 \times 0.2) = 0.5$

As the formula implies it, the coverage score is defined on a $[0, 1]$ interval. As a reminder, the min function is necessary to guarantee a maximum coverage score of 1. Since some controls may overlap, the sum of the mitigations can add up to more than 1. In our approach we consider that providers which do that (implement ineffective security controls) should not be advantaged in comparison to others. Moreover, in our approach, it does not make sense to have a coverage score higher than 1, since such a value would have no signification.

This **coverage score** allows us to compare different providers based on a list of threats. Thus, it can already be used for selecting a provider in some simple use cases. It can be understood as the response capacity of a provider to a given threat. Or on the contrary, to make a parallel with the **vulnerabilities**, it can be considered as the exposition of a provider to a specific threat. In the Definition 14, this analogy will be used to calculate the risk.

We admit that our approach is debatable and could need an adaptation for some use cases. Indeed, one could consider that completely implementing a security control that fully mitigates a threat ($implementation = 1$ and $mitigation = 1$) is sufficient for having a coverage score equal to 1. In this case a probabilistic approach by calculating the *union* of the implemented controls and their mitigations could be interesting, since it would better reflect the intended purpose. In some other cases, one could argue that a calculation with the *intersection* or the *average* could be more interesting. In any case, it is still feasible through our cloud risk assessment model, only the formulas need to be adapted. In this sense we argue that the main contribution of this chapter does not consists in the formulas for calculating the different values, but more in the approach for calculating the risk of different cloud deployment solutions.

4.2.5 Harm: security needs and threat consequences

In this section we define the concept of **harm** (Definition 13).

Definition 13 (Harm) *A rate calculated for a given asset and a given threat. It is obtained by combining the security needs of the asset and the corresponding consequences of the threat. Formally,*

$$\begin{aligned} \textbf{Harm}: & Asset \times Threat \rightarrow Rate \\ & a, t \mapsto Harm(t,a) \\ & = 1 - \prod_{c \in Criterion} \Big(1 - \big(Consequence(t,c) \times Objective(a,c)\big)\Big) \end{aligned} \tag{4.3}$$

The **harm rate** represents the impact of a given threat on a selected asset. It allows us to differentiate the consumer's assets, as all does not necessarily need the same level of security, especially for each threat. As the formula implies it, the harm rate is defined on a $[0, 1]$ interval.

Basically, some threats will be more important for some assets than others, an effect that is typically represented with the **impact** in the classical risk formula (see Section 4.1). In our approach, this concept is reflected in the combination of the **consequence** with the **objective**. It can be understood as the exposition of the asset to a specific threat (to make the analogy with the previously defined **coverage**). For example, assets that do not necessarily need to be continuously *available*, will not be heavily exposed to the *denial of service* threat.

Obviously, in our approach this information can be defined in a more fine-grained fashion than through such a "textual" form. We use a probabilistic approach and calculate the harm rate through the *union* of the asset's objectives and the threat's consequences. In this way, we are sure that when a threat has a maximum severity on a given criterion ($= 1$), and an asset has the maximum need of this criterion ($= 1$), the harm of this threat on this asset will be maximal ($= 1$).

For example, we take an asset a_1 that has a need equal to *usual* ($= 0.5$) in *availability* and a *trusted* ($= 1$) need of *non-repudiation*. Given the *data loss* threat t_1 with a *significant* ($= 1$) consequence on *availability* and a *related* ($= 0.5$) consequence on *non-repudiation*, we get a harm for the threat t_i on the asset a_i of:

$$harm(t_i, a_i) = 1 - \Big(\big(1 - (1 \times 0.5)\big) \times \big(1 - (0.5 \times 1)\big)\Big) = 0.75$$

We admit that this probabilistic approach can be criticized since in some cases another formula, such as the *maximum value* or the *average*, could be more adapted. When taking other risk assessment methods (like the ISO27005 [ISO11]), the impact is generally assessed by taken the maximum value. So in the above case it would be equal to 0.5. However, in those methods the impact (such as the need) is often defined over integer values, which make such a probabilistic approach more difficult to implement and understand for non-technical users. Moreover, we argue that our approach allows to be more precise. Indeed, we consider that an asset that can be adversely affected in terms of multiple security criteria, will be "more" impacted than one that can only be affected by one. Typically, when taking the previous example with an asset a_j with only a *trusted* ($= 1$) need of *non-repudiation*, we get a harm value for the same threat of:

$$harm(t_i, a_j) = 1 - \Big(\big(1 - (1 \times 0)\big) \times \big(1 - (0.5 \times 1)\big)\Big) = 0.5$$

Yet, by taking for example the *maximum value*, these two assets cannot be distinguished in terms of their harm values for this threat. This is why we preferred the probabilistic approach. But once again, we argue that the main contribution of this chapter does not consists in the formulas. One could easily adapt the calculations for another use case, without changing the global approach.

4.2.6 Risk: threat probability

In this section we formalize the final security risk value for a given asset on a given cloud offer and for a specific threat (Definition 14).

Definition 14 (Risk Level) *A level calculated for a given threat, a given asset and a given offer. It is the product of the harm of the threat on the asset and the vulnerability of the offer to this threat. This value can then be weighted by the probability of the threat. The vulnerability is obtained by using the complementary of the offer's coverage. Formally,*

$$\begin{aligned} \textbf{Risk}: & Threat \times Asset \times Offer \rightarrow Level \\ & t, a, o \mapsto Risk(t, a, o) \\ & = \frac{k_h \times Harm(t, a) + k_c \times \Big(1 - Coverage(o, t)\Big)}{k_h + k_c} \times Probability(t) \\ & with\ k_h, k_c \in \mathbb{N} \end{aligned} \tag{4.4}$$

The **threat probability** is a percentage (*i.e.* a value in a [0, 1] interval). It can be used to put the focus more on some threats than on others. Indeed, it is generally accepted that some threats are more likely to happen than others. Indeed, a meteor strike could completely erase a data centre and there are no countermeasures at all that could prevent such an event. However, the probability that it happens is really low. Without taking into account this probability, the risk of this threat would be unnecessarily high.
More concretely, the CSA gives a ranking for their considered cloud threats [CSA13]. This ranking shows that globally, a *data loss* threat is more likely to happen than a *malicious insider* threat. The threat probability of our model allows us to integrate this information in our risk formula.
Moreover, we argue that this information can be adapted for some use cases. Indeed, some businesses are more likely to be the target of *denial of service* attacks than others. Thus, the probability of this threat should be increased. On the contrary, small businesses may ignore some of the cloud threats, since a targeted *data theft* would be too much effort for a potential attacker in comparison to the promising gain.

Our calculation of the **risk level** is similar to the risk of the formula given in Section 2.2.1.1. However, we use a sum to better reflect the independence of the vulnerabilities from the impacts in the cloud context (as the consumer cannot reduce the vulnerabilities of the provider, and the provider cannot be involved for the evaluation of the impact).
One can note that we use the complementary of the coverage to represent the vulnerabilities of the usual risk formula. Theoretically, a provider that fully implements all given security controls would have a coverage of 1, and thus a value for the vulnerabilities of 0. Still, it makes sense since this value would represent the vulnerabilities of the provider regarding a given reference list of security controls. Our approach is not meant to calculate an *actual* risk value (which is probably impossible to undertake), but rather give a reference to compare different cloud providers. This in mind, it makes sense that two providers implementing completely all security controls, would have the exact same risk values. With the **risk level** we are able to compare providers. It classifies the providers towards their risk for deploying a given asset.

We also introduce coefficients in order to weight the formula. Indeed, it can be interesting in some cases to focus more on the impact than on the vulnerability. By default, a weighting of (1,1) reflects the average. The division is not really necessary, but it brings the risk level to a [0, 1] interval, to remain consistent with the rest of the approach. In this way we take a step towards the definition of a "standardized" risk value.

4.3 Running example

In this section we come back to our running example of Section 1.4 and apply our cloud security risk assessment model on it. For that, we selected five candidate cloud providers from the CSA STAR Registry and compared them based on their responses to the CAI-Questionnaire. We remind the example in FIGURE 4.4.

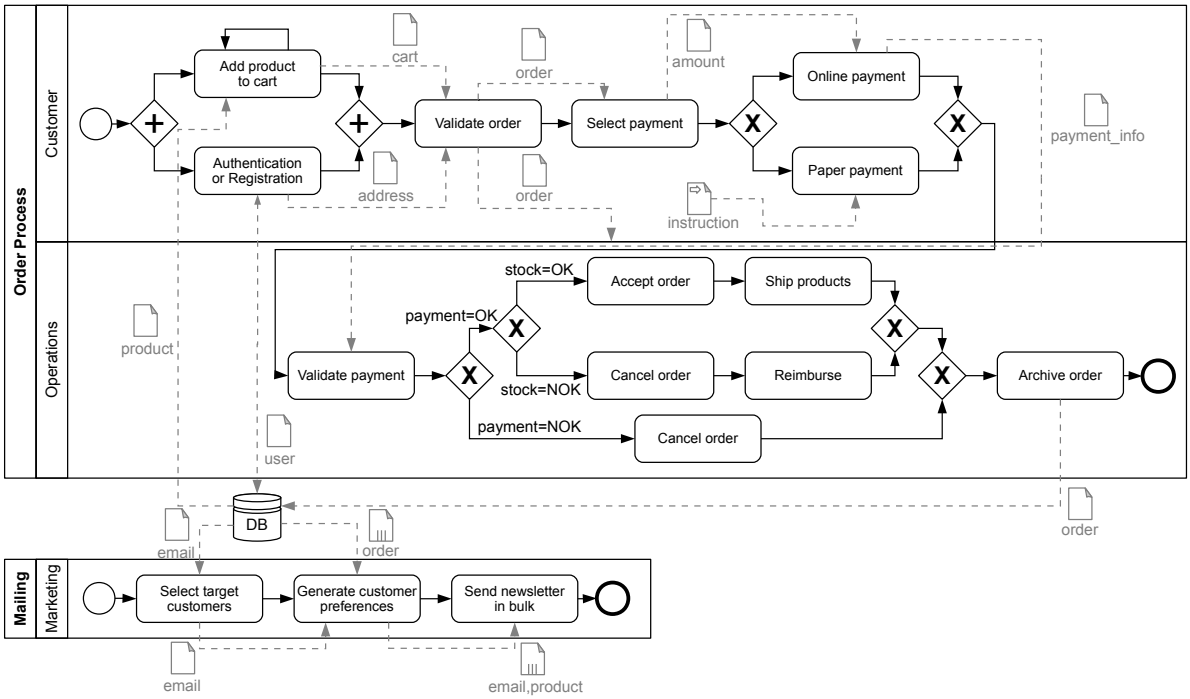


FIGURE 4.4 – Illustrating example - Business processes candidate for being outsourced

4.3.1 Defining the security objectives

First, we take our example processes and annotate them with security objectives in order to be able to assess the impact potential security incidents could have on them. The security needs are expressed on the five CIANA security criterion (*Confidentiality*, *Integrity*, *Availability*, *Non-repudiation* and *Authenticity*) through five levels which are assigned with the values given in TABLE 4.1.

TABLE 4.1 – Levels of security needs and their values

Confidentiality	Integrity	Availability	Non-repudiation	Authenticity	Values
Public	*Losable*	*Sporadic*	*Dubious*	*Anonymous*	0.00
Unclassified	*Alterable*	*Regularly*	*Assumed*	*Known*	0.25
Restricted	*Recoverable*	*Usual*	*Contestable*	*Unsigned*	0.50
Classified	*Correctable*	*Interruptible*	*Evident*	*Legitimate*	0.75
Secret	*Fixed*	*Continuous*	*Irrefutable*	*Genuine*	1.00

We express security needs on the different states of the data objects given in Section 1.4 (so typically on the data associations). TABLE 4.2 shows the values we assigned for our motivating example. The table gives directly the values for the object, without specifying in details the needs of its attributes. However, the needs are given for each attribute of the object and are then aggregated into one need value for the entire object. The need of the object is the maximum need of its attributes. Each

line corresponds to a state of one data object, the state is specified with a ▷ notation that indicates the corresponding data association of the business process.

TABLE 4.2 – Annotations of the security objectives on the running example.

Data states	Confidentiality	Integrity	Availability	Non-repudiation	Authenticity
user $[DB \triangleright AoR]$	*Secret*	*Fixed*	*Interruptible*	*Evident*	*Legitimate*
product $[DB \triangleright AptC]$	*Unclassified*	*Fixed*	*Continuous*	*Irrefutable*	*Genuine*
cart $[AptC \triangleright VO]$	*Restricted*	*Correctable*	*Continuous*	*Contestable*	*Unsigned*
address $[AoR \triangleright VO]$	*Classified*	*Correctable*	*Interruptible*	*Evident*	*Legitimate*
order $[VO \triangleright SeP]$	*Classified*	*Fixed*	*Interruptible*	*Irrefutable*	*Genuine*
order $[VO \triangleright \mathbb{S}]$	*Classified*	*Fixed*	*Regularly*	*Contestable*	*Legitimate*
instruction $[\Omega \triangleright PP]$	*Public*	*Correctable*	*Usual*	*Contestable*	*Genuine*
amount $[SeP \triangleright OP]$	*Classified*	*Fixed*	*Interruptible*	*Irrefutable*	*Genuine*
pay_info $[OP \triangleright VP]$	*Secret*	*Fixed*	*Interruptible*	*Irrefutable*	*Genuine*
order $[ArO \triangleright DB]$	*Classified*	*Recoverable*	*Regularly*	*Assumed*	*Known*
email $[DB \triangleright STC]$	*Restricted*	*Correctable*	*Usual*	*Assumed*	*Known*
order $[DB \triangleright GCP]$	*Classified*	*Recoverable*	*Regularly*	*Contestable*	*Unsigned*
email $[STC \triangleright GCP]$	*Restricted*	*Correctable*	*Usual*	*Assumed*	*Known*
email $[GCP \triangleright SN]$	*Restricted*	*Correctable*	*Usual*	*Assumed*	*Known*
product $[GCP \triangleright SN]$	*Public*	*Recoverable*	*Usual*	*Assumed*	*Known*

Typically, the *user* (state $[DB \triangleright AoR]$) is composed of *{password, email, address, payment_info}*. At this state, the *payment_info* has no need at all, since it is not defined yet. But the *password* has a *confidentiality* need of *secret*, hence this level for the entire *user* object. Once the *payment_info* is defined (state $[OP \triangleright VP]$), the need in *Non-repudiation* and *Authenticity* is higher then at the states before.

TABLE 4.3 – Task-centric security objectives of the running example

Tasks	Confidentiality	Integrity	Availability	Non-repudiation	Authenticity
Order Process					
Auth. or Registr.	*Secret*	*Fixed*	*Interruptible*	*Evident*	*Legitimate*
Add prod. to cart	*Restricted*	*Fixed*	*Continuous*	*Irrefutable*	*Genuine*
Validate order	*Classified*	*Fixed*	*Continuous*	*Irrefutable*	*Genuine*
Select payment	*Classified*	*Fixed*	*Interruptible*	*Irrefutable*	*Genuine*
Online payment	*Secret*	*Fixed*	*Interruptible*	*Irrefutable*	*Genuine*
Paper payment	*Public*	*Correctable*	*Usual*	*Contestable*	*Genuine*
Validate payment	*Secret*	*Fixed*	*Interruptible*	*Irrefutable*	*Genuine*
Accept order	*Classified*	*Fixed*	*Regularly*	*Contestable*	*Legitimate*
Ship products	*Classified*	*Fixed*	*Regularly*	*Contestable*	*Legitimate*
Cancel order (1)	*Classified*	*Fixed*	*Regularly*	*Contestable*	*Legitimate*
Reimburse	*Classified*	*Fixed*	*Regularly*	*Contestable*	*Legitimate*
Cancel order (2)	*Classified*	*Fixed*	*Regularly*	*Contestable*	*Legitimate*
Archive order	*Classified*	*Fixed*	*Regularly*	*Contestable*	*Legitimate*
Mailing Process					
Sel. target cust.	*Restricted*	*Correctable*	*Usual*	*Assumed*	*Known*
Gen. cust. pref.	*Classified*	*Correctable*	*Usual*	*Contestable*	*Unsigned*
Send newsletter	*Restricted*	*Correctable*	*Usual*	*Assumed*	*Known*

With Formula 4.1 given in Section 4.2.2.1, we can generate TABLE 4.3 where each task is assigned to its generated security need. Notice that we work on the initial example, and not the modified one of Section 3.3. Those process transformations should be made once the security risks have been evaluated. Indeed, prior to this assessment, their is no formal way of knowing if the modifications are beneficial or not.

4.3.2 Calculating the harm based on the consequences

For the running example we defined consequences between the nine CSA threats and the five CIANA security criteria. The values are based on the CSA report [CSA13] and are presented in TABLE 4.4. We defined three *Severity* levels and mapped them to the following numerical values: $\{Negligible = 0.0, Related = 0.5, Significant = 1.0\}$. Typically, a *Denial of Service* threat has a *Significant* consequence on the *Availability* of an asset, and a *Negligible* consequence on the other criteria. Whereas *Malicious Insiders* have a *Significant* consequence on all criteria. Similarly, the *Shared Technology Vulnerability* has a *Related* consequence on all criteria because it does not directly target the asset, but rather functions of the operating system (and so indirectly the assets of the user). In some cases there can be significant consequences, but mostly they are confined to a specific part of the system, and thus consequences are less meaningful.

TABLE 4.4 – Consequences of the nine CSA threats on the five CIANA criteria

Threats	**Confidentiality**	**Integrity**	**Availability**	**Non-Repudiation**	**Authenticity**
Data Breaches	*Significant*	*Negligible*	*Negligible*	*Negligible*	*Negligible*
Data Loss	*Negligible*	*Negligible*	*Significant*	*Related*	*Negligible*
Serv. Traffic Hijack.	*Significant*	*Significant*	*Related*	*Related*	*Related*
Ins. Interf. and APIs	*Significant*	*Significant*	*Related*	*Negligible*	*Significant*
Denial of Service	*Negligible*	*Negligible*	*Significant*	*Negligible*	*Negligible*
Malicious insiders	*Significant*	*Significant*	*Significant*	*Significant*	*Significant*
Abuse of Cloud Serv.	*Negligible*	*Related*	*Negligible*	*Related*	*Related*
Insuf. Due Diligence	*Related*	*Related*	*Significant*	*Negligible*	*Negligible*
Shared Techn. Vuln.	*Related*	*Related*	*Related*	*Related*	*Related*

These values seemed to be coherent for our use case and should globally kept this way. However, they can be adapted for other use cases, especially when considering different types of offers or a different granularity for the security needs. For example, for services that provided directly the virtual machines (*cf.* IaaS offers), the *Shared Technology Vulnerability* threat should be considered as having a higher consequence than on SaaS offers. Moreover, it may be interesting to have more than 3 severity levels in some cases, or directly define them with numerical values. In the same sense, the criteria or even the threats can be interchanged with other ones. One can for example define a consequence table between the 35 ENISA risks [ENI09a] and the Parkerian Hexad security criteria [40].

By combining the previously generated task-centric security objectives with these consequences, it is possible to calculate the **harm** of each threat on the process' data attributes. For that we use the formula defined in the previous section (Formula 4.3). It generates the TABLE 4.5 and represents the harm a threat would have on the task if the event happens. Since it is calculated, the harm is a numerical value, but it may also be brought to "textual" levels afterwards the calculations, like: $\{Insignificant = [0, 0.3], Relevant = (0.3, 0.6), Critical = [0.6, 1]\}$.

40. Confidentiality, Possession/Control, Integrity, Authenticity, Availability, Utility

Typically, tasks that handle the same or similar states of data objects, have the same values (*e.g.*, **Accept order**, **Cancel order**, **Ship products**, *etc.*). Intuitively, tasks with the same levels of needs (and thus the same values of harms) will be deployable on the same offers, provided that there are no other constraints forbidding this.

TABLE 4.5 – Harm values of the nine CSA threats on the running example's tasks

	Data Breaches	Data Loss	Serv. Traffic Hij.	Insec. Interf.	Denial of Service	Malicious insiders	Abuse of Cloud Serv.	Insuf. Due Diligence	Shared Techn. Vuln.
Order Process									
Auth. or Registr.	1.00	0.84	1.00	1.00	0.75	1.00	0.80	0.94	0.94
Add prod. to cart	0.50	1.00	1.00	1.00	1.00	1.00	0.88	1.00	0.95
Validate order	0.75	1.00	1.00	1.00	1.00	1.00	0.88	1.00	0.96
Select payment	1.75	0.88	1.00	1.00	0.75	1.00	0.88	0.92	0.95
Online payment	1.00	0.88	1.00	1.00	0.75	1.00	0.88	0.94	0.96
Paper payment	0.00	0.63	0.93	1.00	0.50	1.00	0.77	0.69	0.82
Validate payment	1.00	0.88	1.00	1.00	0.75	1.00	0.88	0.94	0.96
Accept order	0.75	0.44	1.00	1.00	0.25	1.00	0.77	0.77	0.87
Cancel order (1)	0.75	0.44	1.00	1.00	0.25	1.00	0.77	0.77	0.87
Cancel order (2)	0.75	0.44	1.00	1.00	0.25	1.00	0.77	0.77	0.87
Ship products	0.75	0.44	1.00	1.00	0.25	1.00	0.77	0.77	0.87
Reimburse	0.75	0.44	1.00	1.00	0.25	1.00	0.77	0.77	0.87
Archive order	0.75	0.44	1.00	1.00	0.25	1.00	0.77	0.77	0.87
Mailing Process									
Sel. target cust.	0.50	0.56	0.93	0.93	0.50	0.96	0.52	0.77	0.73
Gen. cust. pref.	0.75	0.63	0.97	0.98	0.50	0.99	0.65	0.80	0.84
Send newsletter	0.50	0.56	0.93	0.93	0.50	0.96	0.52	0.77	0.73

4.3.3 Providers coverage scores

For the running example we selected five cloud providers from the CSA STAR Registry [Clo14] which published the security controls they have implemented for their services. The mitigations can be obtained from [CSA13], where the CSA relates some security controls to their nine top cloud security threats. We enhanced these mitigations in order to relate each control to at least one threat. Since these two listings (implementation and mitigations) are very consequent (297 controls), they are not published in this section, but can be found as annex at the end of the manuscript. More current information about the providers can be found online, on the CSA STAR Registry [Clo14].

In accordance with the formula defined in Definition 12, these information lead us to calculate the coverage scores presented in TABLE 4.6. These values must be considered as being subjective. Indeed, different values for the mitigations would lead to different coverage scores. Even the information about the implementation can be interpreted in different ways, since some provider do not publish this information in a binary form. However, we argue that it will only slightly change the scores and that the global ranking to compare the providers will not be significantly impacted. This is why we

are confident that such type of numerical evaluations will emerge in the future and that standardized values will be defined.

TABLE 4.6 – Coverage scores of five providers for the nine CSA cloud threats and their probability

	Softlayer	CloudSigma	FireHost	SHI Intern.	Terremark	Probability
Data breaches	0.36	0.62	0.21	0.54	0.52	0.91
Data loss	0.49	0.59	0.42	0.59	0.67	0.91
Account Hijacking	0.55	0.64	0.46	0.59	0.46	0.87
Insecure interfaces	0.54	0.67	0.53	0.59	0.58	0.90
Denial of service	0.60	0.66	0.53	0.62	0.66	0.81
Malicious insiders	0.49	0.64	0.54	0.60	0.61	0.88
Abuse of Cloud Services	0.58	0.60	0.56	0.54	0.59	0.84
Insufficient Due Diligence	0.53	0.64	0.51	0.60	0.56	0.81
Technology Vulnerabilities	0.49	0.64	0.50	0.55	0.54	0.82

Moreover, TABLE 4.6 gives the probability of each threat. For our example we followed the information given in [Clo14]. These values are based on the answers to a survey (*Is the threat relevant ?*) conducted by the CSA and given by domain experts. But these must be considered as subjective or at least context-dependent. For example, the probability of the *Shared Vulnerability* threat when considering IaaS offers is more important than when comparing PaaS or SaaS offers. The probabilities could also follow the classification of the threats given by the CSA (from the most important to the least important). However, the CSA classifies these threats regarding their "severity", meaning that they already take into account their possible "impact". In our case, the probability reflects only the likelihood of the event (so basically, is it probable that the threat happens or not). In this perspective, these survey values are more meaningful.

4.3.4 Generating the final risk values

Finally, we can calculate the risk values by combining TABLE 4.5 with TABLE 4.6 based on Formula 4.4. As explained in Section 4.1, the result is a three-dimensional matrix where a risk value is defined for each $\{task, offer, threat\}$-tuple. To be able to represent the results in a readable form, we aggregate the values to get only one value for each $\{task, offer\}$-tuple: only the highest threat is retained. Such type of aggregation is very common in risk assessment methods, since the assessment is generally made to filter out the most important risks.

For our running example we considered a weighting of {1, 1}, to equally take into account the **harm** and the **coverage**. The final risk values are shown in TABLE 4.7.

By defining a threshold to specify an acceptable risk value, we can filter out cloud offers that are below a given value. For example, with a threshold of 0.8, the offer provided by **FireHost** would not be selectable for the tasks **Authentication or Registration**, **Online Payment** and **Validate Payment**. Such a threshold can be seen as a security requirement (informational constraint, see Section 3.1.2) given by the consumer.

What is interesting to notice is that the risk is not defined globally, but for each task. Since our approach is based on the idea that processes can be deployed on a multi-cloud environment, it makes

TABLE 4.7 – Risk levels for the tasks of the running example and five providers

	Softlayer	CloudSigma	FireHost	SHI Intern.	Terremark
Order Process					
Auth. or Registr.	0.75	0.63	0.81	0.66	0.67
Add prod. to cart	0.69	0.64	0.72	0.64	0.67
Validate order	0.69	0.64	0.72	0.64	0.67
Select payment	0.66	0.60	0.70	0.63	0.67
Online payment	0.75	0.63	0.81	0.66	0.67
Paper payment	0.66	0.60	0.66	0.63	0.64
Validate payment	0.75	0.63	0.81	0.66	0.67
Accept order	0.66	0.66	0.70	0.63	0.67
Cancel order (1)	0.66	0.66	0.70	0.63	0.67
Cancel order (2)	0.66	0.66	0.70	0.63	0.67
Ship products	0.66	0.66	0.70	0.63	0.67
Reimburse	0.66	0.66	0.70	0.63	0.67
Archive order	0.66	0.66	0.70	0.63	0.67
Mailing Process					
Sel. target cust.	0.65	0.58	0.64	0.60	0.64
Gen. cust. pref.	0.66	0.59	0.70	0.62	0.66
Send newsletter	0.65	0.58	0.64	0.60	0.64

sense to isolate the highest risk values (so basically the most critical tasks) and deploy them on the most secured providers. In the next section we show how a process can be decomposed and deployed on multiple environments. We also present different selection algorithms, since these generated risk values are not the only criterion, they have to be combined with others, in particular with costs.

4.4 Conclusion

In this chapter, we have proposed a technique for assessing security risks of business processes before deploying them in a multi-cloud environment. This technique relies on two main aspects, on the one side business process security needs and on the other side cloud providers guideline conformance. By combining the impact evaluation on the cloud consumer and the vulnerability assessment of the cloud provider, a cloud security broker can help companies to deploy securely their applications on a multi-cloud environment. To illustrate our approach, we have used a running example. We have defined the security needs on the five CIANA security criteria, and we used the CSA security controls to evaluate the risk levels of five cloud providers taken from an industry-recognized registry: STAR [Clo14]. Since our approach is model driven, it can be extended to other sets of controls or criteria.

In short, the points that need to be retained from this chapter are the following:

- the **cloud consumer** defines security **objectives** over his **assets** by using a set of security **criteria** as a reference. He can quantify his objectives with **need** values.
- the **cloud provider** implements security **controls** on his **offers** to reduce the vulnerabilities of

his system and thus prevent security incidents from happening. Such controls are available in guidelines, standards or certifications and should be published for a better transparency.

- the **cloud broker** has the expertise for defining lists of cloud security **threats**. On the one side these threats can be **mitigated** by different security controls implemented by the providers. On the other side, these threats have **consequences** on different security criteria. These both type of information help to define a risk value for each possible combination of asset, threat and offer. The risk is calculated as follows:
  - **objectives** and **consequences** create a **harm** value that is synonym of the impact of a possible cloud security threat.
  - **implementations** and **mitigations** create a **coverage** score that represents the security response of a cloud offer to a given threat.
  - the *risk* is the combination of the **harm** and the **coverage** and can additionally by weighted by a **probability** for the given threat.

Chapter 5

Deployment of a business process on multiple clouds

This chapter is mainly based on the contributions published in [GFG13]. It details how a business process can be broken down in different fragments and the resulting fragments can be deployed in a multi-cloud environment. In this objective, the work in this thesis has been designed and integrated in a larger approach [FDGG14] that is overviewed in Section 5.1. This previous work details how a business process can be automatically partitioned in sub-processes according to different quality of service parameters, security risk being one of them.

Even if this thesis focuses on security aspects of cloud environments, the business reasons to move applications to the cloud are completely different: cost savings, flexibility, scalability, simplified maintenance, *etc.* (see Section 2.1). Therefore, our security risk assessment model cannot be considered as being sufficient for selecting providers as long as it cannot be combined with other parameters. In this perspective, this chapter presents a multi-criteria cloud selection algorithm to find an optimized deployment configuration. Additionally, with a multitude and always expanding set of existing cloud offers, it can rapidly become complicated to evaluate all deployment possibilities. This is also an issue considered in our approach.

5.1 Overview

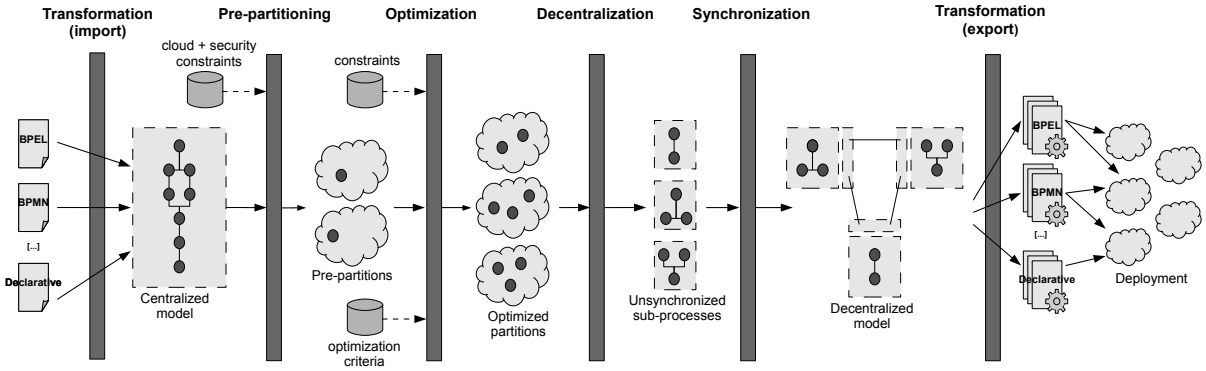


FIGURE 5.1 – Overview of the decentralization approach

The backbone of the approach is an algorithm to split a business process into small partitions.

The algorithm for doing this automatically has been developed prior to this thesis and can be accessed in [FYG09]. This partitioning transforms a centralized process into behaviourally equivalent and distributed sub-processes each of which is related to a set of web services. These partitions are executed independently at distributed locations (clouds) and can be invoked remotely. They directly interact with each other using asynchronous messaging without any centralized control. An overview of the different steps required to achieve such a decentralized business process execution is given in FIGURE 5.1.

First, control and data flow dependencies of the centralized process model specified with a source language (BPEL, BPMN, *etc.*) are analysed and transformed into process graphs (used as a pivot format, see Section 5.3.1). This makes the decentralization approach independent from the specification language.

In the pre-partitioning phase (see Section 5.3.2), tasks are grouped in partitions to verify the constraints defined by designers: typically, one constraint can impose two tasks to execute on different clouds (separation), and another can impose two tasks to execute on the same cloud (co-location) ; tasks can also be grouped based on other kind of constraints, including their requested security level and/or requested role for execution. The types of constraints that can be taken into account at this step have been detailed in Section 3.1.2.

As a result of the previous step, partitions contain a mapping consistent with constraints and available cloud solutions. However, as several candidates can exist for enacting a given task, this first decomposition may not be optimal in terms of costs or other criteria like quality of service. Therefore, the next step (see Section 5.3.3) consists in refining this initial splitting using optimization techniques in order to minimize the communication costs between the derived partitions and maximize the overall quality of service. Our contribution concerning the multi-criteria cloud offer selection is located at this point, since our objectives are different from those of the previous work. In our work we focus on the security risk in a cloud context, whereas the existing optimization part considers parameters such as geolocation of web-services. The main contribution of this section is to study how the risk metrics developed in Chapter 4 can be integrated in this decomposition approach to generate business processes that can be securely deployed in a cloud environment.

In order to re-construct a connected process model, the next two steps, *i.e.* decentralization and synchronization (see Section 5.3.4), allow respectively the wiring of tasks within a same partition, and the interconnection of the derived partitions using a message exchange mechanism. The derived decentralized model consists of several graph models each related to a partition connected by message exchanges.

Finally, each partition model is transformed to the target language (BPEL, BPMN, *etc.*) and deployed in the generated configuration on the selected clouds (see Section 5.3.5).

Before going into the details of the adaptation of the decomposition approach to the cloud context, in the following we present some criteria that can be considered for the cloud offer selection in addition to the security risk presented so far.

5.2 Criteria to evaluate

In this section we present other types of criteria that should be taken into account when selecting cloud offers in combination with security risks. Indeed, minimizing the risk of a deployment configuration is not sufficient for motivating a cloud outsourcing. These criteria are formalized in order to combine them with security risk metrics and to develop an automated and scalable selection approach.

Obviously, our approach is not limited to consider the risk metrics defined in Chapter 4. Our contribution is generic and allows to take into account other type of metrics. Results of risk assessments can be very different, especially regarding the scales on which the risks are evaluated. Sometimes risk levels are defined on an interval of 0 to 20, sometimes on one of 0 to 10. Sometimes, methods simply give a qualitative evaluation of the risk (like *high* or *medium*) and others limit their evaluation to a classification of the risks by their importance. Thus, it becomes very complex to define a generalized approach for minimizing risks while considering other types of parameters.

5.2.1 Costs

Our cost model takes into account three types of costs: *Usage Costs*, *Storage Costs* and *Transfer Costs*. We observed that these three types are the most characteristic of the cloud business model. Indeed, we found out that generally each pricing model can be mapped to these three attributes.

Usage Costs (C_{usg}) correspond to the price for CPU power on the selected cloud offer. Generally, on IaaS cloud offers, the consumer can select which computing power he needs (often directly in terms of GigaHertz [41]). In our case, we have decided to express these costs in Dollars per Gigahertz on a hourly basis, since many providers adopt this type of pricing scheme. Moreover, we annotate each task of our business process model with the need in terms of CPU power per hour, called the *Execution Cost* (C_{exc}). This allows then to calculate the computing power required by each process fragment and deduct the costs for executing a fragment on a given offer.

Storage Costs (C_{str}) correspond to the price of storing space on the selected cloud offer. This is expressed in our model in Dollars per Gigabyte per hour. Each data object of the process is annotated with its estimated size, the *Size Cost* (C_{sze}). Additionally, we need the retention period (*RP*) for the data object, *i.e.* how long the data object has to be stored. *E.g.*, if we consider that the data of each process instance is stored for one month, we can easily calculate the needed storage space for each instance. To extrapolate it to a monthly value, we need the number of process instances executed over one month.

Transfer Costs (C_{trs}) correspond to the amount of incoming and outgoing data between the cloud and external services. Generally, cloud providers bill their consumers according to the transferred Gigabytes of data, independently of their direction (in- or outgoing). These costs are usually expressed in Dollars per Gigabyte. We calculate this amount using the *Size Cost* of the data exchanged between the process fragments. If the tasks of the process use external web services, these must also be taken into account.

Note that some tasks may not be executed for some instances (particularly tasks that are in an *OR-branching*). To get a more precise approximation, the tasks of the process model can be annotated with an execution probability (P_{exc}). This probability can be higher than 1 in the case of loops, where the same task is executed multiple times. This probability also impacts the other costs, since messages may not be sent, or data objects may not be created.

41. Amazon proposes another type of measure, Elastic Compute Unit (ECU)

Definition 15 (Costs) *For a given deployment configuration of a process (i.e. the mapping of its tasks, its data object and its messages to different cloud offers), the costs of executing one instance of this process is defined as follows:*

$$\begin{aligned} Costs(instance) = & \sum_{t \in tasks} P_{exc}(t) \times C_{exc}(t) \times C_{usg}(o_t) \\ & + \sum_{d \in data} P_{exc}(t_d) \times RP(d) \times C_{sze}(d) \times C_{str}(o_d) \\ & + \sum_{m \in msgs} P_{exc}(t_m) \times C_{sze}(m) \times \left(C_{trs}(o_{m/in}) + C_{trs}(o_{m/out})\right) \\ \text{with } & o_t, \text{ the offer where } t \text{ is deployed} \\ & o_d, \text{ the offer where } d \text{ is stored} \\ & o_{m/in}, \text{ the offer where } m \text{ comes in} \\ & o_{m/out}, \text{ the offer where } m \text{ comes out} \\ & t_d, \text{ the task that creates } d \\ & t_m, \text{ the task that sends } m \end{aligned} \tag{5.1}$$

Optimally, the costs should be defined per instance, since "pay-per-use" is one of the key characteristics of cloud computing. However, different other reference models can be selected, like costs per day, per month or per year. This often depends on the use case and the compared cloud offers. Typically, when deploying a process execution engine over an IaaS cloud offer, it will be really difficult to evaluate the costs of one instance, since the provider will charge the consumer for its global hourly or monthly use. And our model does not take into account the costs of executing a process execution engine. However, on a PaaS offer, that already implements the process execution engine, it will be easier for the cloud provider to bill his consumers according to the executed instances. In this sense, we do not limit our cost model to any of those references: our process model is annotated with the requirements of one instance, which can then be extrapolated to the costs over a certain period. To do this we need the information of how many instances are expected to be executed during the period (Nb_{ins}/p).

5.2.2 Quality of Service

Quality of Service can include a lot of different parameters and can be considered from very different perspectives. And when it comes to measuring it, thus defining metrics to compare different services, it gets even more difficult.

There are several commercial services proposing solutions to compare existing cloud offers. We do not give an exhaustive list, but different example to show the different characteristics that can be considered to fit into a *Quality of Service* criteria.

- CloudScreener[42], proposes a score on 100 points based on a combination of performance, security, flexibility and price. The score takes also into account the consumer's need to prioritize the characteristics. The information and the evaluation algorithm are periodically updated. The comparison is limited to IaaS offers.
- SoftwareInsider[43], proposes to select cloud offers based on user ratings. It is a website that is specialized in comparing software solutions based on the reviews given by consumers. Thus,

42. `http://www.cloudscreener.com/en/`
43. `http://cloud-computing.softwareinsider.com/`

it is not really a *measured quality*, but more a *perceived quality of service*. However, it can still be an interesting factor to take into account.

- HostAdvice [44], proposes a mix between expert and customer reviews. It takes into account five different criteria: *Reliability*, *Pricing*, *User-friendly*, *Support* and *Features*. Additionally, this service allows customers of these offers to post their reviews in form of *pros* and *cons*.
- TopTenReviews [45], compares features available on different cloud services. The number of compared offers is really limited, but the service calculates an interesting score for the selected providers. Features are considered as more or less important, and the more features are implemented, the better the provider's score is.

Among the scientific community, there are also existing contributions that propose solutions to assess cloud providers. The authors of [AJG+15] made a systematic mapping study to survey current research in cloud QoS approaches. They identified IaaS as the main focus area, and state that validation research type represents the majority of the reviewed articles.
The authors of [BH14] provide a list of metrics that can be taken into account for measuring QoS. For each feature, like *Communication*, *Memory*, *Elasticity*, *Efficiency*, *etc.*, they give a list of metrics that fit into this definition (like *Packet Loss Frequency*, *Boot Time*, *etc.*).
Noticeable work is the proposal of Baliyan *et al.* [BK13] that present an assessment model based on fuzzy logic. In opposition to other approaches requiring numerical values, here the approach can work with other types of information similar to our risk annotations (*High, Medium, Low, etc.*).
Becker *et al.* [BLB15] derive metrics for scalability, elasticity and efficiency properties of cloud services using the goal question metric (GQM) method. They define six different units to measure those metrics and compare provider services numerically.
Some other work, as that presented in [BYOG13], propose to evaluate the quality of service by considering two criteria: the cost and the execution time. The approach propose fairness metrics for selecting cloud providers in order to deploy business applications.

Even if a common compromise for evaluating Quality of Service seems to be quite far of being achieved, a summary we draw from our state of the art is that the usual approach is to achieve a numerical score on which providers can be compared. Thus, for the following we assume in our selection approach that QoS can be brought to a score, typically a percentage where 100% is the best achievable value.

Once again, we do not aim at defining or providing a realistic metric for evaluating the QoS of existing cloud providers, but we want to emphasize that it is another important selection criteria that should be taken into account. It is meant to motivate our multi-criteria selection approach, to find a good configuration for deploying a business process on clouds, by considering as many valuable information as possible.

5.2.3 Complexity

As already explained previously, our approach relies on a process deployment on multiple cloud environments. A process is not necessarily deployed on a single location, but can be fragmented over different services, some of them can even be on-premises ("hybrid cloud").

However, such a configuration can create important interoperability issues, since all services do not rely on the same technologies, standards, languages, protocols, *etc.*. Adaptations, or other situation-specific configuration may require additional expenses. Also, some tasks such as *contract*

44. http://hostadvice.com/
45. http://cloud-services-review.toptenreviews.com/

negotiation, *service subscription* or *configuration* must be carried out multiple times, making the outsourcing very difficult in comparison to a single cloud deployment. Moreover, once deployed, the more cloud providers are involved in the deployment, the more probable it gets that one will fail or suspend his services, threatening the entire process execution.

These different points lead to argue that there are some configurations that are more "complex" than others. And this must be taken into account for the final configuration selection. Since such criteria (like the costs of a migration, or the time needed for developing an API-specific interface) are difficult to evaluate, we define a very simple metric based on the number of different services involved in a configuration (Nb_{srv}) and the number of tasks of the process to deploy (Nb_{tsk}). Our formula for the complexity of a configuration is defined as follows:

Definition 16 (Complexity) *The complexity of a deployment configuration for a business process is given by the following formula:*

$$Complexity(conf) = \frac{Nb_{srv}(conf)}{Nb_{tsk}(conf)} \tag{5.2}$$

The denominator is important to make the value of the complexity more generic. It is easier to accept many involved services for large processes than for processes including few tasks. In this way, two configurations with a different number of tasks can be compared upon their complexity.

An interesting point is that some could also see this value otherwise. One could argue that a high value of *complexity* is synonym for a good protection of the *know-how* of the process. Indeed, if the process is fragmented over lots of different locations, it will be more difficult for an attacker to gather all the different pieces together to rebuild the original process and understand it. Thus, how this value is used, is completely open to the user of our approach and can be used differently in other use cases.

5.2.4 Functional requirements

In software engineering, functional requirements are often described through functional specification based on UML diagrams. However, such types of representation are not formalized enough to be useful for an automated selection. Since it is not part of the core contribution of this thesis, we do not explore in details the possibilities of the modelling and formalization of functional requirements on business processes. We rather rely on existing work and integrate it in our approach.

To note is the generic eSourcing Capability Model for Client Organizations (eSCM-CL, [KH07]), defined by the ITSqc at the Carnegie Mellon University. It is the complementary to the capability model for service providers (eSCM-SP). It is designed to give the possibility to service providers to express their capabilities and map these to the customers requirements. The model is organized through a life-cycle, different capability areas (like *Knowledge Management*, *People Management*, *etc.*), capability levels and best practices.

Specific to business processes, a modelling of functional requirements has been proposed in [FJ12]. The authors propose a meta-model for a formal specification of functional requirements, implemented as a BPMN extension. The model is intended for a dynamic service selection.

In our approach, we assume that for each task a pre-selected set of consistent services is defined. So basically, the functional requirements are given in the form of constraints that define if a task can be deployed on a given cloud offer or not. Therefore we give the following function:

Definition 17 (Functional requirements)

$$\textbf{Deployable}: Task \times Offer \rightarrow Boolean$$

$$t, o \mapsto Deployable(t, o) = \begin{cases} True, \text{if } o \text{ can execute } t \\ False, otherwise \end{cases} \quad (5.3)$$

As said before, it is not the core of the thesis to specify how cloud offers can be selected based on functional requirements. This is a completely different topic that requires additional work. This work focuses on non-functional requirements, especially security requirements expressed in form of risk levels. However, since functional requirements are still an important aspect (even the most important one), we integrate it with such a formula that is generic enough to take into account a lot of different approaches. We assume that symmetrically, in a service selection approach exclusively based on functional requirements, our security requirements could be integrated similarly. Except that we translate functional requirements in form of non-negotiable constraints, whereas non-functional requirements (such as our risk levels) can ultimately be questioned and it can be decided to not respect them (typically if the costs are too high).

5.2.5 Other non-functional requirements

Even if security is one of the most important non-functional requirements, there are also a lot of other requirements that should be taken into account when selecting cloud offers. Similar to the functional requirements, we do not formalize these in a proper way, since it is not part of the main objective of our work. However, it is important to know that some other criteria exist and can be interesting in some cases.

Especially in cloud environments, some criteria become more important in comparison to an on-premises configuration. For example, the location of the provider can be an important argument to select a cloud offer or not. To minimize the risks related to possible legal differences between jurisdictions, it can be interesting to prefer the selection of cloud providers one one country. Even if this risk can already be considered in our cloud security risk assessment model, it can be more explicitly expressed through a criteria to optimize.

Another aspect can be response-time, that can have an interesting added value. Some cloud offers provide (and guarantee) better response-times than others. Here again, this information can be integrated in the cloud security risk assessment model defined at the previous chapter, but cannot be explicitly taken into account.

Other criteria to mention when outsourcing to a cloud environment can be the following: documentation, environmental protection, open source, portability, usability, *etc.*.

Typically, we distinguish between two types of criteria: those which are considered as constraints and have to be respected, and those which are considered as optimization criteria. The first type leads to the exclusion of configurations, since a configuration that does not respect these constraints cannot be selected. The second type helps us to define when a configuration is better than another one, and helps us in the end to select the “best” configuration. Functional requirements are basically constraints (first type). All other criteria can be considered as parameters to optimize (second type), but can in some use cases be considered differently. Typically, legal requirements may lead to enforce the location for some data types, and cloud offers relying on foreign data-centres must be excluded. But those requirements are taken into account with our logical, organizational and informational constraints defined in Section 3.1.2.

Our goal here is not to make an exhaustive list of criteria that have to be taken into account in our cloud offer selection approach, but rather to underscore the importance to consider other parameters than security and costs. The *Cost*, *Quality of Service* and *Complexity* can obviously be mapped into the non-functional requirements. One could even argue that these criteria "to optimize" could be grouped into one big Quality of Service category. Indeed, some approaches or solutions shown previously integrate directly security information or financial costs into a generic Quality of Service criterion in order to separate them from the "constraints". Here, we present these criteria separately to put the focus on them, since we think that these are extremely important to consider in our approach.

5.3 Decomposition and deployment approach

The deployment of a business process in a multi-cloud environment needs specific adaptations to enact adequately to its original purpose. Here we present the adaptation of an existing approach (see [FYG09]) to decompose a business process in fragments, each fragment being intended for one cloud offer. Those fragments can then be executed in a decentralized fashion. To fully automate the decomposition and deployment, the approach also takes into account the selection of the cloud offers. Overall, the approach consists in five different steps: **transformation**, **pre-partitioning**, **optimization**, **decentralisation and synchronisation** and **deployment**.

5.3.1 Transformation

The first step consists in transforming one business process modelling notation into another to support a language independent approach. Thus, the module takes as input a model in his specific modelling notation and transforms it into an internal directed graph structure. In this graph structure, nodes are either tasks, events or control patterns (*e.g.*, *parallel*, *choice*, *repeat*, *etc.*), and edges are the process flows (*e.g.*, *sequence*, *default*, *conditional*, *etc.*). All decomposition operations are performed over this representation, to finally output the decomposed process, in either the original notation, or another one (this in the limits of the supported notations by our internal representation). Currently supported notations by the developed tool (see [GFG13]) are *BPMN* and *BPEL*, but others can also be integrated such as *YAWL* [46] or *EPC* [47].

The module is built on 2 parts:

- At the front-end, an **import** part provides methods to parse a given file and to generate the equivalent internal graph. Currently the supported formats are BPMN and JSON [48].
- At the back-end, an **export** part generates the executable files for the given graph. Currently BPMN (2.0), BPEL and dotGraph [49] are supported. For some notations, adaptations on the graph have to be done. In BPEL for example, *multi-send* and *multi-receive* patterns are not supported and are translated into parallel branches of *send* and *receive* tasks.

5.3.2 Pre-partitioning

The pre-partitioning phase distributes tasks in *partitions* to ensure the respect of constraints, typically three types of security constraints as described in Section 3.1.2. As a reminder,

46. Yet Another Workflow Language, `http://www.yawlfoundation.org/`
47. Event-driven Process Chains, `https://en.wikipedia.org/wiki/Event-driven_process_chain`
48. JavaScript Object Notation, http://www.json.org
49. Graph Visualization Software, http://www.graphviz.org/

- **logical constraints**, given at the level of the whole process. *I.e.*:
  - split the process into a minimum (or maximum) number of fragments
  - systematically separate the logic from the data
  - group tasks or data related to a type of activity
- **organizational constraints**, given at the level of the process' tasks. *I.e.*:
  - separate tasks into two different fragments
  - co-locate tasks in the same fragment
  - impose that one task should remain on-premises
- **information constraints**, taking into account the cloud offers characteristics. *I.e.*:
  - ban a given cloud for a given task, data or fragment
  - impose a given cloud for a given task, data or fragment
  - impose a level of security (typically a risk level) for a task

The goal of this step is to generate a first set of partitions that respects all given constraints. Since all presented constraints are not *orthogonal* (*i.e.* some can be contradicting), this step ensures that there actually exists a possible solution. If not, the approach immediately stops at this point to request the user to change the given constraints.

The pre-partitioning phase also calculates the minimum and maximum number of possible partitions according to the constraints. These information are an input parameter for the optimization algorithm.

5.3.3 Optimized selection

The aim of the optimization step is to produce partitions such that the previously described criteria are either maximized or minimized. Basically, it is intended to find the optimal solution regarding the considered criteria to evaluate.

Unfortunately, this problem can be considered as a multi-objectives optimization problem, which is not straightforward to solve. Indeed, there is no "universal" method to find the optimal solution of such a problem, primarily because of the difficulty to define the *optimal solution*. The term of *Pareto optimality* is used to describe a state where no better solution can be found without making at least one criterion worse, but *Pareto solutions* are rarely limited to one single state.

Additionally, we face here a *NP-hard* problem, were N tasks have to be mapped to P cloud providers. More precisely, it consists in a Quadratic Assignment Problem (QAP) [BePP98]. Since there is no known algorithm to resolve such type of problem in a polynomial time and no "standard" way of solving it, we propose our own way of finding an optimized assignment, given the different criteria to evaluate.

5.3.3.1 Considering multiple criteria

First, we tackle the problem of considering multiple criteria. In general, multi-criteria optimization methods can be mapped into four different categories as defined by Hwang & Masud [HM79]:

- *no-preference strategies*, where a neutral compromise is identified, without any interaction of a *decision maker*
- *a priori strategies*, where a *decision maker* defines his preferences before searching for the corresponding solution

- *a posteriori strategies*, where a *decision maker* selects his preferred solution among a set of Pareto solutions
- *interactive strategies*, where the *decision maker* iteratively searches for the most preferred solution

In the following we will present three different and most commonly used methods to find a "good" assignment of business process tasks to cloud offers. The fourth method consists of our own proposition according to the considered criteria.

Linear scalarization The easiest and most intuitive way of solving a multi-criteria problem is to transform it into a single-criterion optimization problem. One way to do that is to define for all k criteria named c_i, a weight w_i. Thus the problem of finding the best solution among the set of X possibilities, is reduced to find:

Definition 18 (Linear scalarization)

$$\min_{x \in X} \left(\sum_{i=1}^{k} w_i \times c_i(x) \right) \tag{5.4}$$

While this method seems applicable to all use cases, it brings a major problem concerning the units (metrics) of the scalar value. Indeed, all criteria are expressed with different units (dollars, seconds, scores, *etc.*). Therefore, the variations of each criteria between different solutions can be very problematic for defining the right weightings. Moreover, when changing the unit of one criteria (for example euros instead of dollars), the formula could lead to a different "best" solution. Thus, even if the approach can be generalized, the weightings must be adapted to the context of the considered scenario.

Especially when considering security risk, this approach can be very arduous, since there are no standardized risk metrics. Thus, the weightings would have to be redefined for each different use case. Moreover, defining weightings for comparing information like *high* or *low* with quantitative cost values is not intuitive, and could potentially lead to arbitrary decisions.

ϵ-constraint Another way to transform the multi-criteria problem into a single-criterion one, is to select only one criterion to optimize and transform the other ones into constraints. In this case, the optimization is performed on only one criterion, and a threshold ϵ_i is defined for all other criteria that has to be respected. With the same notation as previously, we get the following:

Definition 19 (ϵ-constraint)

$$\begin{aligned} &\min_{x \in X} \big(c_j(x) \big) \\ &\quad \textit{with } c_i(x) \leq \epsilon_i \textit{ for } i \in \{0, ..., k\} \setminus j \end{aligned} \tag{5.5}$$

This approach is the most widely used one, since it is the best translation of the real world conditions. Often, the criterion to optimize are the costs, and for all others, a threshold is defined (such as it is done for the risk in most of the existing security risk assessment methodologies). Therefore, the "best" solution that is found, is the cheapest one, given that the *risk*, the *QoS*, the *complexity*, *etc.* are above (or below) a given value.

In comparison to the linear scalarization approach, where the problem is the adequate definition of the weightings, here it is easier to define the thresholds. Indeed, they are defined independently,

and do not have a mutual influence (in the linear scalarization, increasing one weighting is synonym of decreasing the importance of all others). However, the problem of this method is that it cannot really be considered as a multi-criteria approach, since only one criterion is optimized. Especially in the context of risk, such an approach tends to "hide" acceptable solutions: maybe a risk value slightly below the threshold, could be accepted if the overall costs are drastically lower.

Pareto subset Here we do not detail a specific method, but merely a principle, that of the *a posteriori* strategy. It consists in selecting a set (or a subset) of the Pareto solutions and present them to the *decision maker* who can select the preferred configuration among the proposed ones. Such approaches usually express a dominance function that specifies when a solution is better than another. In the case of the Pareto-dominance, all criteria must be better to have a domination of one of the solutions. The approach consists then in eliminating all solutions that are Pareto-dominated by another one, the remaining ones form the set that can be presented to the decision maker.

In our context, such an approach makes a lot of sense, since the *decision maker* entity can correspond to our cloud broker. Therefore it is not unthinkable to define an approach where different possible deployment configurations are presented by the cloud broker to the cloud consumer, who can then make his choice. Similar to the risk, it is also possible that no acceptable solution is found, and that the cloud deployment is aborted.

The main drawback of a Pareto set is that it can be really large. In this case, this approach is completely uninteresting, since the number of possible solutions to review by the cloud broker is way too large to do it manually. Whereas the goal of our approach is to support the broker with a selection approach that can be automated and significantly lower the required manual actions for the reviewing process.

Hybrid approach To combine the benefits and remove the drawbacks of each of the previously mentioned approaches, we propose a hybrid method to find the best deployment solutions among a set of possible configurations. Our approach can be characterized with the following three points:

- the weightings of the **linear scalarization** strategy are interesting because they allow to easily define which criteria are decisive and which one are less important. Therefore, our approach is based on weightings that are assigned to the criteria. However, the linearisation combines units that do not fit together and creates a value that has no real significance. This drawback is avoided in our approach because it does not associate different values with each other, but works only on the weightings themselves.
- the thresholds of the ϵ**-constraint** strategy are interesting because they can easily be defined, independently one of each other. However, we want to optimize all criteria and not only work on one and use the thresholds to exclude solutions. Therefore, our approach is a real multi-criteria optimization algorithm and thresholds are only used to compare solutions one-by-one.
- the **a posteriori** strategy, that proposes a set of solutions to a *decision maker* is a good idea in our context, especially because of the existence of the cloud broker. Thus, our approach does not propose one solution considered as "the best one", but rather aims at generating an acceptable set of "good" solutions. These solutions are a subset of the *Pareto solutions*.

Our approach can be formalized as follows, with the same notation as previously, and S being our generated set of solutions:

Definition 20 (Hybrid approach)

$$\forall x \in S, \nexists y \in S \text{ such that } y \succ x, \text{ with}$$
$$x \succ y \Leftrightarrow \begin{cases} \sum_{i=0}^{k} w_i[c_i(x) >_i c_i(y)] > T, & \text{with } T \text{ the global threshold} \\ \forall i \in \{1,...,k\}, c_i(x) >_i c_i(y) +_i t_i, & \text{with } t_i \text{ the threshold of } c_i \end{cases} \quad (5.6)$$

Nota Bene: the $>_i$ sign does not mean *greater than*, but rather *better than*. The same applies to the $+_i$ sign, which means *augmented of*, rather than *plus*. Indeed, some criteria are inverted, *ie.* *QoS* is better when the value is higher, but for *Costs* or *Risk* it is better when the value is lower. This is implicit in our approach, but must be specified for each newly created criterion before it can be automatically considered in our algorithm. This is why we talk of an optimization approach and not a minimization problem.

Textually, our set of solutions contains only configurations that do not dominate each other, while the domination of x over y is given by the two conditions:

- the sum of the weights of the criteria where x is better than y must be greater than a global threshold value (generally defined at 50% of the total of the weightings).
- for all criteria, the value of y cannot be better than that of x of more than the threshold for that criterion (it represents the difference for which the first assertion is reconsidered).

Example To better illustrate our approach, we propose the example given in TABLE 5.1. Three possible configurations are given with their values for four criteria (*Costs*, *QoS*, *Complexity* and *Risk*). The two columns on the right give respectively the weighting for the criterion, and the threshold. As a global threshold we define 50 (since the sum of the weightings equals to 100).

TABLE 5.1 – Example for the hybrid approach

Criteria	Sign	Config. 1	Config. 2	Config. 3	Weightings	Thresholds
Costs ($)	$-$ (<)	*150.00*	*140.00*	*170.00*	*60*	*50.00*
QoS	$+$ (>)	*8/10*	*6/10*	*7/10*	*10*	*3/10*
Complexity	$-$ (<)	*0.76*	*0.81*	*0.66*	*10*	*0.20*
Risk	$-$ (<)	*2/15*	*5/15*	*7/15*	*20*	*2/15*

Now we compare all configurations one by one to see which one have to be retained:

- Config. 1 $\succ$ Config. 3, because 150.00$ is better than 170.00$ and $60 > (T = 50)$. Moreover:
  - 8/10 of QoS is better than 4/10 of QoS
  - 0.76 of Complexity is better than $0.66 + 0.20 = 0.86$ of Complexity
  - 2/15 of Risk is better than 7/15 of Risk
- Config. 2 $\succ$ Config. 3, because 140.00$ is better than 170.00$ and $60 > (T = 50)$. Moreover:
  - 6/10 of QoS is better than $7/10 - 3/10 = 4/10$ of QoS
  - 0.81 of Complexity is better than 0.66 of Complexity
  - 5/15 of Risk is better than 7/15 of Risk
- Config. 2 $\not\succ$ Config. 1, because 140.00$ is better than 150.00$ and $60 > (T = 50)$. However:
  - 6/10 of QoS is better than $8/10 - 3/10 = 5/10$ of QoS

- 0.81 of Complexity is better than $0.76 + 0.20 = 0.96$ of Complexity
- $5/15$ of Risk is **not** better than $2/15 + 2/15 = 4/15$ of Risk

Our approach allows us to exclude the Config. 3 as a viable solution, but concerning Config. 1 and Config. 2 no decision can be taken, it is the role of the **decision maker** to do that. Indeed, the *Risk* value of Config. 1 is *so much better* than the *Risk* of Config. 2 that it allows to reconsider the gain in terms of cost of Config. 2.

This multi-criteria selection approach can be considered as an independent block of our general methodology for deploying business processes in multi-cloud environment. Indeed, one could choose another multi-criteria approach for excluding non-viable configurations, or even apply simply one of the methods presented previously.

5.3.3.2 Heuristics for the QAP

The second problem is the NP-hardness of the assignment of the process' tasks to cloud offers. In order to build a complete approach, we have to consider that there are many possible providers that are available for fulfilling a given task of a process. Moreover, processes can also be very large and consist of a considerable number of tasks. Therefore, even if it seems exaggerated for our introduced examples so far, ignoring a quadratic problem would greatly affect the scalability of our approach.

The same problem has already been tackled in the previous work [FDGG14] with a heuristic approach to prevent the exploration of all possible solutions. The considered criteria were the quality of service, the communication costs between each process fragment and the distance between the selected services.

However, the optimization was performed with a linear scalarization comparison, which we consider as not adequate within the context of risk considerations. The existing approach was based on a *Tabu search* [GL97] algorithm built upon an initial solution generated using a *Greedy algorithm* [MF02]. The Greedy algorithm builds an initial elite solution regarding the different criteria by assigning the best available offer to each constrained task. Based on the Greedy elite solution, the Tabu search tries to improve iteratively the partitioning by moving some tasks in other partitions or even in new partitions.

Our adaptation of this approach is basically the same idea of combining a Greedy elite solution with a Tabu search algorithm. However, we do not use the linear scalarization for comparing our different criteria and we generate a set of possible configurations that are considered as being "good".

- the *Greedy solution* is given as output of the pre-partitioning phase. The algorithm incrementally places each task at an acceptable location, while respecting the constraints given over the process (*co-location*, *separation*, *etc.*). At this point, the process is not considered as a whole, but only local optimal points are explored. Thus, each task is assigned to the offer/fragment that fits the most. To do that, we only focus on the *Usage Costs* and do not consider the other criteria. Several reasons motivate this design decision:
  - the objective is to find a possible configuration that is not "too bad", thus it is not of paramount importance to have an advanced evaluation method.
  - the Tabu search necessitates an existing solution to enhance, thus we cannot generate a set of multiple possible configurations
  - the costs are generally the most decisive criterion, thus it prevents us from defining a complex formula as required for the linear scalarization.

Once again, it is possible that no solution is found, in this case constraints must be re-defined, new cloud offers must be added, or the whole cloud outsourcing is aborted.

- the *Tabu search* algorithm tries to improve this initial solution by moving tasks to other partitions. As input to this algorithm a *depth* value is given, that indicates how "deep" the exploration should go (so basically the number of moves). The higher this value is, the more configurations are evaluated, and once this number reached, the algorithm stops. At this point the hybrid multi-criteria approach is used to compare the obtained configurations. Thus, some of the evaluated configurations are excluded, while the other ones are retained. The output of the algorithm is than the set of "good" configurations from which the cloud broker can select the most appropriate one.

Once again, this algorithm is a heuristic approach, which means that it does not guarantee the optimal solution, but only gives an optimized solution. Here again, we do not argue that our solution is the only one that should be used. For small problems it is probably better to evaluate all possible solutions, because it will be possible to find the "best" solution in an affordable time. Our goal here is to show that our general approach is modular and can be easily combined with other blocks, approaches or methods than those presented here.

At this step, tasks are assigned to partitions, and partitions are assigned to clouds. However, tasks are not linked to each other with neither control- nor data-flows. Note that multiple partitions can be assigned to the same cloud (particularly if only a logical separation constraint is set on two tasks).

5.3.4 Decentralization and synchronization

After obtaining the different partitions (each partition is a set of tasks), the distinct sub-processes have to be generated, *i.e.* the tasks of the partition have to be wired, respecting the initial semantic of the global process. To do that, the approach uses *Transitive Control Dependency Table* (TCDT). The construction and the usage of those table is explained in detail in [FYG09].

At this step, the sub-processes are generated and correspond to independent task compositions accordingly to the original centralized process. However, to recreate the exact execution sequence as specified originally, these sub-processes must be synchronized. First, the correct sequencing of the tasks (control flow) must be enforced. Second, it must be guaranteed that the data needed by a task is available for the task needing it when executed (data flow).

Control flow synchronization To guarantee the semantic equivalence with the original centralized process, synchronization primitives are inserted into each sub-process. They are of two types: *send* and *receive*. By exchanging synchronization messages, each sub-process ensures that its tasks are executed in the right order. A *receive* primitive blocks the execution of the sub-process until the proper message, sent by a remote partition, is received. A *send* notifies a remote partition that an activity has ended and that it can start the succeeding ones. Once again, details about the insertion of such synchronization controls are given in [FYG09].

Important to note is that in case of *OR-branches* where one whole branch is located in another partition, *fictive* tasks are inserted. These tasks are necessary to allow the partition to bypass a task. It corresponds to the case when the branch is executed which is not located on the current partition. So, there must be a path to continue the execution of the process while not executing the task of the other branch. Such *fictive* tasks are simply empty tasks that do nothing.

Data flow synchronization A data dependency represents a relation between two tasks such as: the data produced by the first one is consumed by the second. We can notice two problems: (1)

send data to a task which will not be executed and (2) wait for data from a task which will not be executed. Details for these data synchronization patterns are given in [GFG13].

The unconsumed synchronization and data messages are deleted after expiration or after the termination of the corresponding partition instance. FIGURE 5.2 illustrates the result of the synchronization applied to our running example.

At this step, we choose the initiator sub-process and a variable to identify the global process instance. The initiator is in charge of launching other sub-processes, initiating the first task and transmitting to the other partitions the process identifier.

Advanced patterns Some specific patterns that can be encountered in a business process model cannot be managed with the previous decentralization and synchronization patterns. One of them is the *loop*, that allows for a specific block of the process model to be executed multiple times. The number of loops can be given at design time, or at run time. In both cases, it has to be considered separately for the decomposition. The idea for partitioning a process model with a loop, is to identify the loop block and to treat it as a separate process. Therefore, first the loop block is decomposed, tasks are assigned to partitions, the control flow of the partitions are re-constructed and the different partitions of the block are synchronized. Only after that, the remainder of the process model is decentralized, by considering the already generated partitions. The different blocks of the initial loop can than be considered as a single task, and the remaining task can than be added before or after that one.

Nota Bene: an important condition for this approach to work is that the initial process model must be *well-structured*. Basically, a *well-structured* model is a process where to each opening *(X)OR-split gateway* corresponds a closing *(X)OR-join gateway*. The same goes for *AND-split* and *AND-join gateways*. In addition, the blocks between these gateways cannot overlap. The authors of [PGBD12] formalize such type of models and show that in theory it is not a problem, since non-structured process models can be transformed into structured models.

5.3.5 Transformation (output) and deployment

Once the decentralized model has been generated, it has to be transformed to an executable format. In the case of BPEL processes, WSDL files are generated to describe the interface of each sub process. Additionally a deployment descriptor in form of an XML file and the corresponding BPEL processes are generated. To easily transform the graph structure into BPEL notation, the tool uses the *jbpt* [50] package provided by the University of Potsdam to represent the process models as RPST [VVK08] (Refined Process Structure Trees). As explained previously, multi-send and multi-receive patterns are transformed into parallel flows of receive and send primitives. The BPEL file's content can be separated in three different sections:

- in the first section we can find the own interface description file, the remote WSDL files and these of the other sub-processes. These files are necessary to understand how to interact with the other processes.
- in the second section we can find the *PartnerLinks*, the *Variables*, and the *CorrelationSets*, necessary to correctly communicate with the remote processes. This section also defines the synchronization messages which have to be sent.
- the last section is the process itself and specifies the workflow of the sub-process.

50. Business Process Technologies 4 Java, `https://code.google.com/p/jbpt/`

The WSDL file defines the entry points of the process: one for starting the process, and the other to allow it to communicate with the remote processes. These endpoints correspond to the *send* and *receive* synchronization tasks inserted at the previous step. The identifier for the global process instance is defined by the initiator sub-process, it is needed to identify the correlation set (so that the remote sub-processes know to which instance an incoming message corresponds).

These files can then be directly deployed on the corresponding cloud services. In the case of existing deployment APIs (like they are defined for the Apache ODE process execution engine [51]), the deployment can even be done automatically and remotely.

An interesting point with this decomposed processes, is that each cloud sees only the interfaces of the remote fragments, and not the content of the sub-processes. In terms of *know-how preservation* or other *privacy* objectives it is a major gain. Indeed, none of the participants can have a complete view of the original process.

5.4 Running example

In this section we apply our decomposition approach to our running example of Section 1.4. This section also illustrates the optimization algorithm and discusses it.

5.4.1 Annotating the process model

First, we annotate our example with the information needed to evaluate the criteria presented in Section 5.2. So basically the attributes necessary for calculating the costs, which are: the *Execution Costs* (C_{exc}) of the tasks and the *Size Cost* (C_{sze}) of the data elements (with its required retention period *RP*). These values are given in TABLE 5.2 and TABLE 5.3.

TABLE 5.2 – Data annotations for evaluating storage and transfer costs on the running example.

	user				**order**				**product**			**cart**
Data objects	credentials	email	address	payment_info	user	address	products	status	reference	description	price	products
C_{sze} in KB	2	1	3	4	6	3	50	1	1	3	1	30
RP in Days	1825	1825	1825	30	365	365	365	1825	1825	1825	1825	15

Additionally to these data objects, we also have to include the size of the synchronization messages. These messages are very lightweight, since they only include the identifier of the executing instance, but must also be taken into account for evaluating the total **Transfer Costs**. Thus, we assume in our example that the size of the synchronization message is equal to 1KB.

5.4.2 Cloud offers information

Then, we need information about the pre-selected cloud offers. For this example we took the same five cloud providers of Section 4.3. TABLE 5.4 gives the different cost values for these providers. Those values were approximated based on the values found on the providers websites. Indeed, providers do not always directly indicate these values, but rather values for one month, or pre-defined

51. Orchestration Director Engine, `http://ode.apache.org`

TABLE 5.3 – Task annotations for evaluating execution costs on the running example.

	Order Process												Mailing		
Tasks	Auth. or Registr.	Add prod. to cart	Validate order	Select payment	Online payment	Paper payment	Validate payment	Accept order	Cancel order	Ship products	Reimburse	Archive order	Sel. target cust.	Gen. cust. pref.	Send Newsletter
C_{exc} in GHz/h	0.2	0.2	0.2	0.1	0.3	0.1	0.2	0.1	0.1	0.1	0.2	0.3	0.3	0.6	0.6

packages. But since offerings are subject to frequent changes (and especially their prices), the values may have varied a lot between the time of the benchmarking, the editing of the manuscript and its publishing. Additionally, we include a Quality of Service score taken from HostAdvice (if the provider is referenced, otherwise the score is arbitrary). Here also, these values cannot be considered as real values, they are only intended for an illustrative purpose. In this work, it is not in our intention to compare real cloud providers but more to illustrate our approach with values not too far away from real use cases.

TABLE 5.4 – Information about five pre-selected cloud offers

	Usage Cost ($/GHz/h)	**Storage Cost** ($/GB/d)	**Transfer Cost** ($/GB)	**QoS Score** (/10)
Softlayer	0.028	0.003	0.10	9.6
CloudSigma AG	0.019	0.006	0.06	7.6
FireHost	0.036	0.093	0.50	8.5
SHI Int., Corp.	0.016	0.010	0.01	6.4
Terremark	0.005	0.008	0.17	5.6

5.4.3 Optimized cloud offer selection

Once the information of the offers and the process are defined, our selection algorithm is executed. Since the number of available offers and the number of tasks to deploy are manageable, we do not use our multi-criteria algorithm or the heuristic approach. These will be explained in detail in Section 6.1.2. The objective here is to show that the approach can also work with other optimization and selection methods.

We decide to compare the overall QoS through the average value of the QoS of all selected offers. Basically, we calculate the mean value of the QoS score for the different selected providers, weighted by the number of tasks they execute. Thus, the more tasks are deployed on a provider, the more its QoS score will be important in comparison to the other providers.

We carry out three different runs to get different types of configurations that a cloud broker can propose to a cloud consumer. The results of these runs are illustrated in TABLE 5.5. To simplify the example we only show the results for deploying the **Order Process**. The cross indicates where the task is deployed.

- The **First run** is a simple Dijkstra algorithm to find the "cheapest" deployment solution. Risk and QoS are completely disregarded. Therefore, this configuration is the most interesting one

TABLE 5.5 – Different possible deployment configurations

	First run					**Second run**					**Third run**				
	Softlayer	**CloudSigma**	**FireHost**	**SHI Intern.**	**Terremark**	**Softlayer**	**CloudSigma**	**FireHost**	**SHI Intern.**	**Terremark**	**Softlayer**	**CloudSigma**	**FireHost**	**SHI Intern.**	**Terremark**
Auth. or Registr.	x						x					x			
Add prod. to cart	x						x					x			
Validate order	x						x					x			
Select payment	x						x					x			
Online payment	x						x					x			
Paper payment	x					x								x	
Validate payment	x						x					x			
Accept order					x		x							x	
Cancel order (1)					x		x							x	
Cancel order (2)					x		x							x	
Ship products					x		x							x	
Reimburse					x		x							x	
Archive order					x		x							x	
Risk					**0.75**					**0.66**					**0.64**
Cost ($/instance)					**104.68**					**198.38**					**223.93**
QoS (average score)					**7.60**					**7.95**					**6.96**

in terms of cost, however the risk value is pretty bad in comparison to the other deployment possibilities. The average QoS score seems to be acceptable.

- The **Second run** is the same algorithm, however with a new constraint as input: we give a threshold value for the risk of 0.7. This has the consequence of excluding all offers where the risk of deploying a task on them is above 0.7. The optimization is still working on the cost in order to find a "cheap" solution given this constraint. This has obviously an non-negligible consequence on the costs, since we can notice that costs are almost two times higher than for the first run. The QoS score is slightly improved, a lucky coincidence, since this value is disregarded for this run.

- The **Third run** is made to find the least "risky" solution. So basically the algorithm tries to optimize the risk value of the overall configuration. Since there are different configurations that can have the same risk value (something very unlikely when considering costs), we use the costs as a second value to select between the different configurations. This means that there are other configurations with the same risk value, but that are more expensive. As expected, the costs increase. However, we notice that the average QoS score is significantly lower than for the two previous configurations. Once again, QoS was not considered by the algorithm.

As explained previously, the broker can propose these three configuration to the cloud consumer. The consumer can then decide which of these configurations he will select according to its preferences (typically is he willing to pay for the difference of risk between the different options). In the following we will work with the configuration obtained in the **Third run**.

5.4.4 Partitioning and deployment

Once the final configuration (mapping of tasks to cloud offers) is selected, the partitioning algorithm is executed. First the decentralized processes are re-built, and then the different fragments are synchronized by adding synchronisation tasks to the process. The resulting partitioned process is shown in FIGURE 5.2.

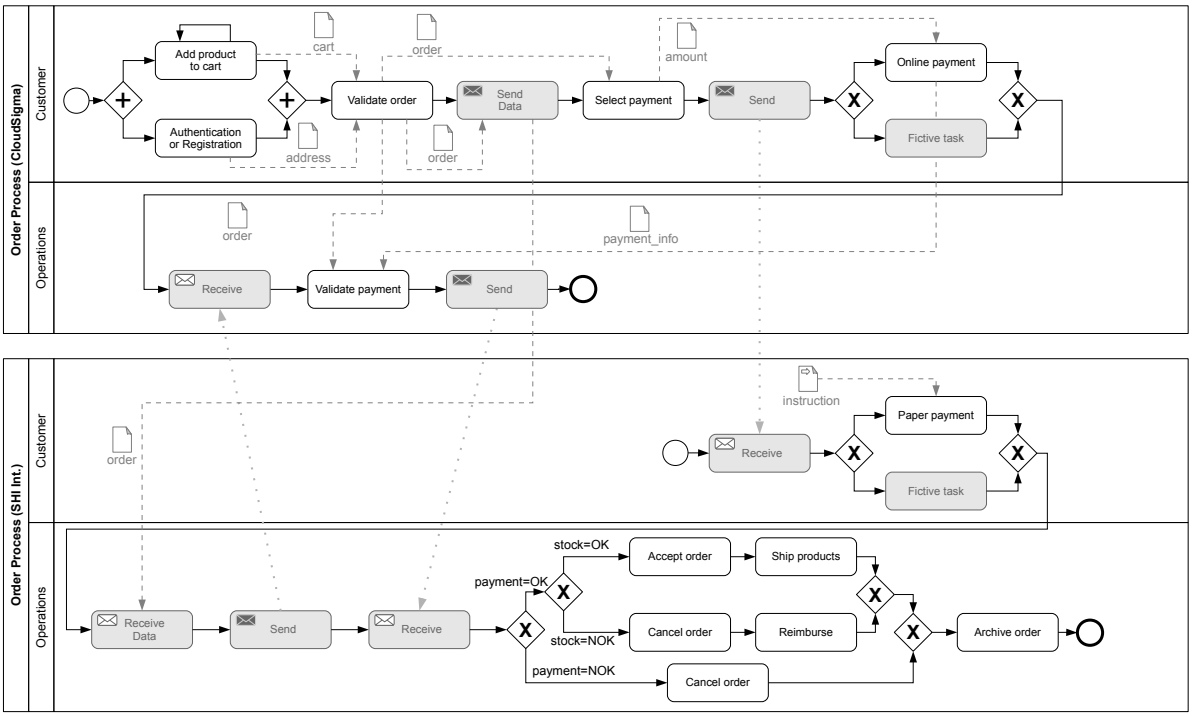


FIGURE 5.2 – Illustrating example - Partitioned business process for a deployment on two clouds

FIGURE 5.2 shows the newly added tasks in grey, to put the focus on them. There are three kinds of new tasks:

- First, there are the **Send** and **Receive** tasks required for synchronizing the control flow. Since it is scattered on multiple locations, it is necessary to notify the other fragments when some tasks are finished in order to continue the process flow as originally intended. In the illustrating example, the first fragment notifies the second one that the **Validate payment** task has been completed. Thus, the second fragment can start the execution of the remaining tasks.
- Second, there are the **Send Data** and **Receive Data** tasks. These are required for the data flow. Indeed, some data objects must be shared between the different fragments and are exchanged through the same messaging mechanism than for the synchronization. In the illustrating example, the tasks of the second fragment need the *order* object that is created by the **Validate order** task in first fragment. Thus, a task is created to send this data object to the remote fragment. The **Receive Data** task redistributes this data object to the other tasks (not shown in the figure for a better readability).
- Finally, there are the **Fictive** tasks. These can be added to the control flow in the case of an exclusive *OR-branching*. Indeed, some branches may be "empty" when all tasks of this branch are located on another fragment. Thus, when this branch is activated, the current process has nothing to do. However, the BPMN notation does not allow a direct sequencing between two

gateways. To prevent such a non-compliance to the standard, we add an "empty" task to that branch. So, these **Fictive** tasks have the purpose of creating a path where the tasks of the other branch are not activated. They have an execution time equal to 0 and do not affect the execution of the process in any way. In the illustrating example, the **Online payment** and the **Paper payment** tasks are separated, which creates the need of adding such type of tasks.

Important to note is that these additional messages created between the different fragments are taken into account for calculating the costs of a configuration, in particular the *Transfer Costs*.

5.5 Conclusion

In this section we detailed how to take into account the risk metrics defined in Chapter 4 with other parameters. In addition we explained how a business process can be automatically fragmented into separate partitions and can than be deployed on multiple clouds. To do that we use an existing approach presented in [FYG09] and enhance it with our multi-criteria optimization approach. The partitioning approach is then illustrated on the running example.

In short, the points that need to be retained from this chapter are the following:

- a business process can be automatically **fragmented** in sub-processes to deploy it in a decentralized fashion on multiple cloud environments. The fragments communicate in a peer-to-peer fashion to recreate the control flow of the original process.
- the main business reasons to move applications to the cloud, like costs, flexibility or scalability are generally not easily reconcilable with security. This is why effective **multi-criteria** approaches need to be developed to compare cloud offers without disadvantaging security.
- the optimized assignment of business process' tasks to cloud offers is a quadratic assignment problem. Therefore, we propose an approach based on **heuristics** to find a suitable configuration for deploying a business process on multiple clouds.
- the proposed multi-criteria method is a **hybrid** approach that combines different existing methods to compare fairly possible deployment configurations:
  - parameters that are considered in the comparison are **weighted**, similar to the linear-scalarization approach.
  - a **global threshold** defines when a configuration dominates another.
  - **local thresholds** are defined for each parameter in order to specify from which value it becomes interesting to reconsider the weighting.
  - the result is a set of possible configurations, to allow the cloud consumer to decide **a posteriori** which one he selects.

Chapter 6

Implementation and validation

This chapter is intended to demonstrate the feasibility and the usefulness of our global approach. First, we describe the tools we developed during this thesis to support our approach. Second, we present a use case in access control where we implemented our security risk assessment model to show that it can easily be implemented in other domains than business processes. Finally, we discuss the approach through a real use case that was carried out during this thesis. We show how our risk assessment model has been useful for selecting a cloud service between different available offers.

6.1 Tool support

Our tool consists in three different modules that respond to different functions of our approach. The first one (Section 6.1.1) allows the assessment of cloud security risks when deploying assets to cloud offers. It is not exclusively dedicated for business processes, even if it is intended for. The second one (Section 6.1.2) is our selection algorithm that can combine multiple criteria to evaluate different possible configurations. This module has been externalized in order to replace it (or select it on-the-fly) when other selection algorithm are more adapted. The last one (Section 6.1.3) is the module for automatically deploying processes (or process fragments) to cloud providers. It is based on a web service architecture to manage the providers and the deployed processes.

6.1.1 Risk assessment

Our risk assessment approach was implemented in a web tool as a proof of concept [52]. It is developed in JavaScript and uses multiple frameworks to support our different features. To allow the user of our tool to model how he wants to assess the risk, we use Blockly [53], an open-source library for building visual programming editors. To maintain consistency between the underlying data model and the interface we use KnockoutJS [54] and JQuery [55]. In the following, we present our risk assessment tool in three stages: (i) model construction, (ii) security needs definition, (iii) risk evaluation for each provider.

52. The tool is available online at: `http://elio.goettelmann.fr/projects/cloudra_v2/`
53. `https://developers.google.com/blockly/`
54. `http://knockoutjs.com/`
55. `https://jquery.com/`

6.1.1.1 Model construction

Or tool is based on the concepts defined in our cloud security risk assessment model (see Chapter 4). Thus, the tool allows to instantiate each of these concepts in several ways and link them to the other concepts. In this way, we allow the user to build his own model, with the parameters and criteria he wants to consider. However, the tool enforces at any point the compliance of the built model to our generic cloud security risk assessment model. In this sense, the tool is adaptable regarding the attributes to use to evaluate the risk.

For example, to define the **security objectives** of his assets, the user can choose the five CIANA criteria as a reference. He can also limit them to three (CIA), use other existing criteria (like STRIDE, or the Parkerian hexad) or even define his own criteria. Sometimes, this can be useful when regulations impose a given reference to consider (in Luxembourg, the financial sector uses ROLF: Reputation, Operational, Legal and Financial). In our tool, the user can select between these different blocks that instantiate the concept of **security objectives**.

For each of these criteria, levels can than be defined to specify the security need. By default, there are two existing levels: either the criterion is needed, or not. In FIGURE 6.1 we define the security needs on the three CIA attributes: *Confidentiality* and *Integrity* are defined on 3 levels, whereas *Availability* uses a slider with 10 levels (that are not textually defined). A value in the interval [0,1] is assigned automatically to each of these levels according to the selected combination type. Typically, the existing combinations are:

- **normalized**, the lowest and the highest levels of all criteria have an equal value. For the levels and criteria of FIGURE 6.1 it would be:

	0.00	0.11	0.22	0.33	0.44	0.50	0.55	0.66	0.77	0.88	1.00
Conf.	*Public*	-	-	-	-	*Restricted*	-	-	-	-	*Secret*
Integ.	*Passable*	-	-	-	-	*Alterable*	-	-	-	-	*Fixed*
Avail.	$Level_0$	$Level_1$	$Level_2$	$Level_3$	$Level_4$	-	$Level_5$	$Level_6$	$Level_7$	$Level_8$	$Level_9$

- **minimized**, the lowest levels have all an equal value (0.00) and the highest level of all the criteria has the highest value (1.00). For the levels and criteria of FIGURE 6.1 it would be:

	0.00	0.11	0.22	0.33	0.44	0.55	0.66	0.77	0.88	1.00
Conf.	*Public*	*Restricted*	*Secret*	-	-	-	-	-	-	
Integ.	*Passable*	*Alterable*	*Fixed*	-	-	-	-	-	-	
Avail.	$Level_0$	$Level_1$	$Level_2$	$Level_3$	$Level_4$	$Level_5$	$Level_6$	$Level_7$	$Level_8$	$Level_9$

- **maximized**, the highest levels have all an equal value (1.00) and the lowest level of all the criteria has the lowest value (0.00). For the levels and criteria of FIGURE 6.1 it would be:

	0.00	0.11	0.22	0.33	0.44	0.55	0.66	0.77	0.88	1.00
Conf.	-	-	-	-	-	-	-	*Public*	*Restricted*	*Secret*
Integ.	-	-	-	-	-	-	-	*Passable*	*Alterable*	*Fixed*
Avail.	$Level_0$	$Level_1$	$Level_2$	$Level_3$	$Level_4$	$Level_5$	$Level_6$	$Level_7$	$Level_8$	$Level_9$

- **centred**, the mid levels of all criteria have all an equal value of 0.50. For the levels and criteria of FIGURE 6.1 it would be:

	0.00	0.11	0.22	0.33	0.44	0.50	0.55	0.66	0.77	0.88	1.00
Conf.	-	-	-	-	*Public*	*Restricted*	*Secret*	-	-	-	-
Integ.	-	-	-	-	*Passable*	*Alterable*	*Fixed*	-	-	-	-
Avail.	$Level_0$	$Level_1$	$Level_2$	$Level_3$	$Level_4$	-	$Level_5$	$Level_6$	$Level_7$	$Level_8$	$Level_9$

Note that some values are not assigned to a level for some criteria. This means that this value does not have any representation. Indeed, it can make sense in some cases to evaluate the *Confidentiality* on three levels, and the *Availability* on more. However, this means that some levels may have no equivalent on the scale of another criterion.

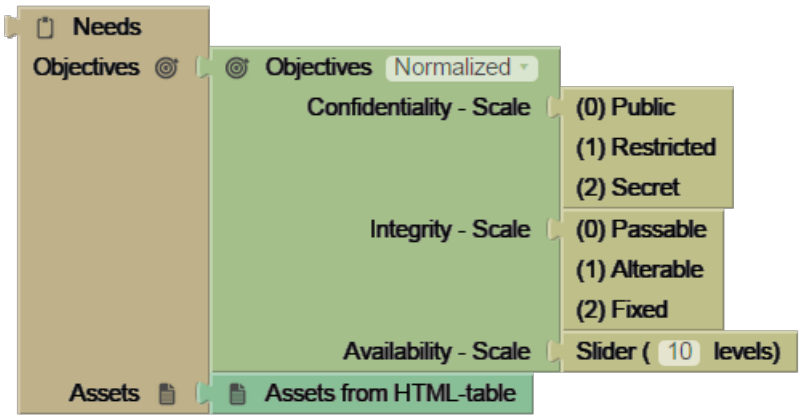


FIGURE 6.1 – Defining security objectives: criteria and their levels

The **assets** can be linked to these criteria through different blocks. Either they are defined manually through a simple HTML-table (as it is illustrated in FIGURE 6.1) or they can be generated from a linked JSON or BPMN business process (as explained in Chapter 4).

For the **consequences**, the user can select the type of **threats** that should be associated to the security criteria. Either the nine CSA threats, or the ENISA threats and even manually defined threats can be given as basis to the consequences. As a reminder, **consequences** link threats to criteria, thus if a *Denial of Service* affects rather the *Availability* than the *Confidentiality* of an asset. Here also, levels can be defined for each criterion (*i.e.* the severity, see Definition 11). The values of these severities are defined in exact the same way as for the security needs: the user can select between a normalized, a minimized, a maximized or a centred combination (always on a interval of $[0,1]$).

The **harms** are then defined through the aggregation of the needs and the consequences. As a reminder, the harm is a value that represents the negative effect the realization of a threat would have on an asset. Thus, it is defined for a *{threat, asset}*-tuple. How the harm is calculated is also an option that the user can parametrize. Since the needs and the consequences are defined on intervals of $[0,1]$, we propose the following aggregation strategies:

- **union**, as given in Formula 4.3 in Definition 13. It represents the sum of the effects, meaning that as soon as there is a maximum need of one criterion, and a threat that maximally affects this criterion, the harm is maximal. The according formula is:

$$1 - \prod_{c \in Criterion} \Big(1 - \Big(Consequence(t,c) \times Objective(a,c)\Big)\Big) \tag{6.1}$$

- **maximum**, represents the maximum negative effect that can occur, independently from the criterion. This is often used in risk assessment methods to consider the maximum possible impact of a potential threat. The according formula is:

$$\max_{c \in Criterion} \Big(Consequence(t,c) \times Objective(a,c)\Big) \tag{6.2}$$

- **average**, represents the average negative effect. This can be useful when considering the impact of a potential threat as a mean value (that will happen in general). The according formula is:

$$\frac{\sum_{c \in Criterion} \Big(Consequence(t,c) \times Objective(a,c) \Big)}{Card(Criterion)} \tag{6.3}$$

At the same time, the user can build its model for calculating the **coverage** scores of selected providers. In exactly the same way as for building the model for the harm, the coverage can be a combination of the **implementations** and the **mitigations**. The central part of the coverage are the security **controls** that are given as input for the mitigations (to relate them to cloud **threats**) and for the implementations (to indicate on which **offers** they are applied). Similarly to the security criteria, these information can be refined through different levels. *E.g.* implementations can be given as binary values (yes or no), pre-defined levels or even sliders. The same goes for the mitigations.

However, the idea behind the blocks for building the model is that each part is modular. Therefore, a block that gives directly a list of providers with their coverage scores can also be selected by the user (such a block can take as input a pre-generated JSON file). Or one could propose another method to evaluate the coverage score of providers and develop the adapted block.

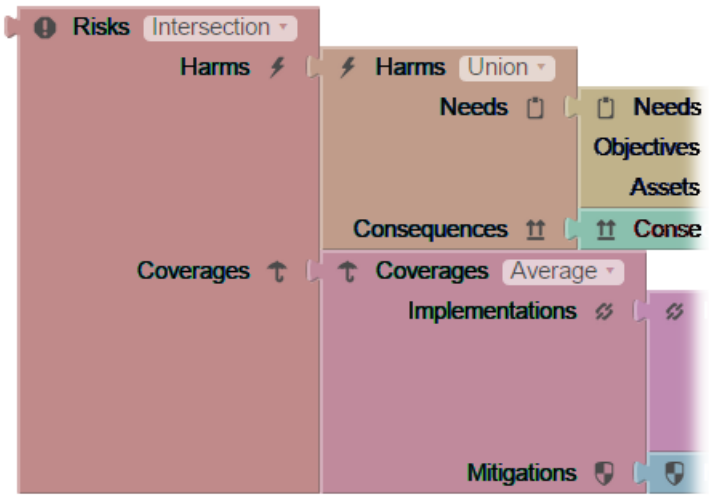


FIGURE 6.2 – Calculating the risk

The building of the complete model leads to configuring how the risk is calculated. For example, the user can choose the risk as the intersection between the harm and the coverage (as in FIGURE 6.2) or as the union, or the average value. This risk block gives as output a three dimensional matrix that relates to each *{threat, asset, offer}*-tuple a numeric risk value.

6.1.1.2 Security needs definition

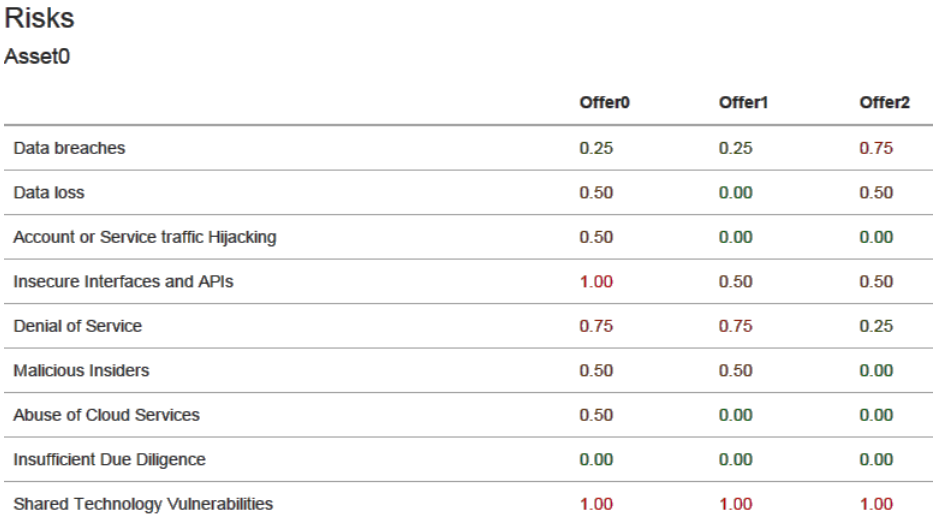
The interface is generated during the construction of the model. Each block has a view where the information that it requires can be entered. For example, the list of assets and their respective needs are given on a specific view. An example is shown in FIGURE 6.3: the needs of *Confidentiality* and *Integrity* are select boxes with three possible values, whereas the need in *Availability* is defined through a slider. This is defined by the definition of the model as explained previously.

FIGURE 6.3 – Defining security needs

Consequences can also be configured in this way, by relating cloud threats to security criteria, accordingly to the blocks selected at the model construction. Similarly, the user can indicate the security controls that offer implement, the threats that these controls mitigate, *etc.*. Note that the interface is always generated according to the blocks selected and the information given in the model. The interface adapts in real-time to the input model.

6.1.1.3 Risk evaluation for each provider

In our tool, the final risk value is also given on a separate view. An example is shown in FIGURE 6.4. For each *{threat, asset, offer}*-tuple, the risk is given, and coloured according to its value (gradually from red for a high risk to green for a low risk). Additional filters can refine this view, for example filtering offers above a given threshold, or only showing the maximum risk value and not detailing it for each threat. Such filters can help to more easily identify the offers that have to be excluded, or the assets that need a strengthened protection.



	Offer0	Offer1	Offer2
Data breaches	0.25	0.25	0.75
Data loss	0.50	0.00	0.50
Account or Service traffic Hijacking	0.50	0.00	0.00
Insecure Interfaces and APIs	1.00	0.50	0.50
Denial of Service	0.75	0.75	0.25
Malicious Insiders	0.50	0.50	0.00
Abuse of Cloud Services	0.50	0.00	0.00
Insufficient Due Diligence	0.00	0.00	0.00
Shared Technology Vulnerabilities	1.00	1.00	1.00

FIGURE 6.4 – Risk values as shown in the web tool

FIGURE 6.5 shows our complete model-based cloud security risk assessment tool. The assessment model, built by the different blocks, is assembled in the drawing area shown in the figure. Blocks are dragged from the menu at the left. The tabs at the top allow to switch between the different views and entering the needed information such as: assets, security needs, consequences and so on.

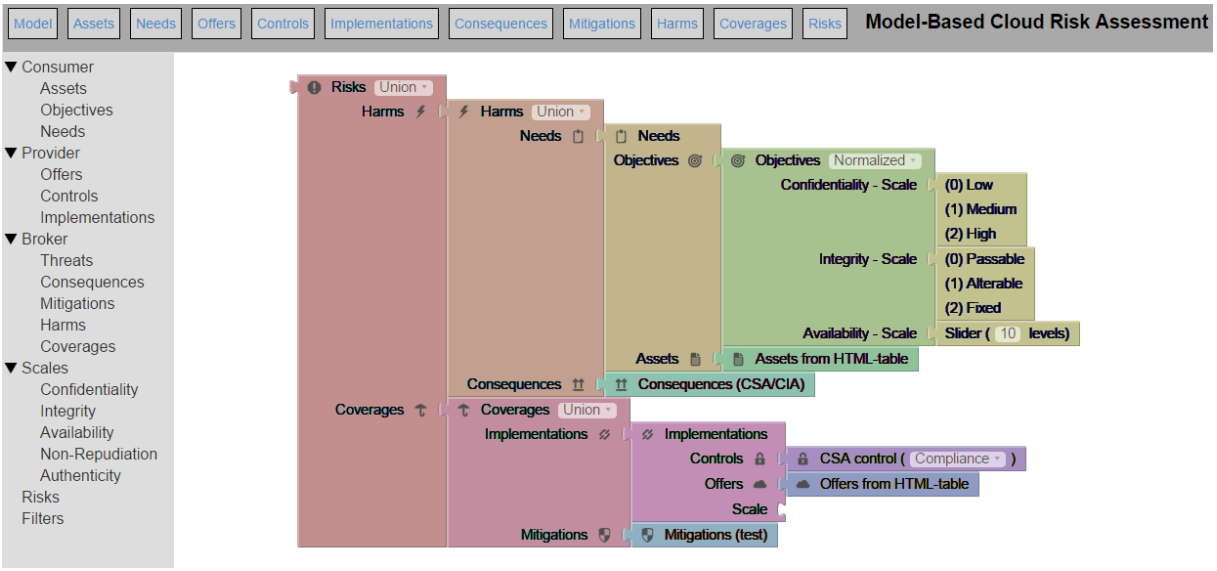


FIGURE 6.5 – Model-based cloud security risk assessment tool

Our tool is developed to support as much flexibility as possible through the definition of independent blocks. The tool can be extended to include other types of blocks that support other risk assessment methods. New types of aggregation or combination methods can also be added to the tool. The main objective of this tool is to demonstrate the feasibility of our cloud security risk assessment approach and implement its model. The dynamic aspect of the tool (on-the-fly generation of the interface) is intended to easily test which type of configuration would be the most adapted for a specific use case. Therefore, this tool can be used to perform a complete cloud risk assessment, but should probably need some improvements (especially regarding performance) to support larger use cases.

6.1.2 Optimized selection

The second part of our implementation is the optimization algorithm, that performs a multi-criteria selection among a set of cloud offers. Before presenting the results of our algorithm on different configuration settings, we introduce the details of our algorithm.

As a reminder, our multi-criteria approach is defined as follows:

- for each criteria we define a weighting, that represents the importance of that criterion. When comparing two solutions we calculate the “sum of the weightings” where the first solution is better than the second one.
- a global threshold defines a value above which the “sum of weightings” determines that solution is better than another one. Generally this value is equal to 50% of the sum of all weightings.
- for each criteria we define a threshold, that corresponds to a difference for which the previous assertion is reconsidered. *E.g.*, a solution that is cheaper, simpler and has a better QoS does not dominate another solution if the second one has a “way better” risk value (the difference of the two risk values is greater than the threshold).
- the result is a set of solutions that do not dominate each other.

6.1.2.1 Algorithm

We implemented the selection based on the hybrid multi-criteria domination property in the form of an algorithm defined in Algorithm 1.

Algorithm 1: Multi-criteria comparison

```
   Input: S;                                              // Possible configurations
   Output: Res;                                           // Good configurations
1  Res_i ← {} ;                                           // Intermediary set
2  foreach s_i ∈ S do
3      foreach s_j ∈ S \ {s_i} do
4          if s_j >_P s_i then                            // s_j Pareto-dominates s_i
5              S ← S \ {s_i}
6          else if s_i ∈ Res and s_i >_H s_j then         // s_i Hybrid-dominates s_j
7              S ← S \ {s_j} ;
8              Res_i ← Res_i ∪ {s_j} ;
9          else if s_j >_H s_i and s_j ∈ (Res ∪ Res_i) then   // s_j Hybrid-dominates s_i
10             Res_i ← Res_i ∪ {s_i} ;
11         else
12             Res ← Res ∪ {s_i} ;
13         end
14     end
15 end
```

Our algorithm takes advantage of the fact that a Pareto-dominated configuration will always be Hybrid-dominated by the same solution. Thus, configurations that are Pareto-dominated can be directly excluded from our final set (Line 5, s_i can no longer be added to *Res*). A configuration that is Hybrid-dominated by another one is obviously also excluded from the final solution set (Line 7). However, such a solution can help to exclude other solutions at a later point and is therefore added to an intermediary set (Res_i). A configuration that does not dominate and that is not dominated by another configuration, can directly be added to the final set of solutions (Line 12).

6.1.2.2 Results

We conducted our evaluations in form of a simulation with four different criteria: *Cost*, *Risk*, *QoS* and *Complexity*. For each test, we generated randomly 100 sets containing between 10 and 1,000 configurations to compare. A configuration consists in a quadruplet of values (one for each criterion, *Cost*, *Risk*, *QoS* and *Complexity*). Each test generated the values of these criteria randomly in a given interval, either with a uniform or a normal distribution. As a reminder, this means that:

- In a **uniform distribution**, all values are equally probable. *E.g.*, in the interval [10,1000] the probability that the costs of a configuration are equal to 10 is the same than that of being equal to 100 or 1000. *I.e.* the criteria values of all possible configurations of a given set are uniformly distributed.
- In a **normal distribution**, all values are distributed along a *bell curve* [56]. This means that the values are not equally probable. In our case, the mean value of the interval is way more probable than one of the extremities.

56. https://en.wikipedia.org/wiki/Normal_distribution

We argue that this is an important factor to test, since one domination constraint is the difference between two configurations (the local threshold). Thus, intuitively, the distribution of configurations (are they generally close to each other or not) should impact the efficiency of our approach.

We tested four different weighting settings and four different threshold settings (TABLE 6.1), resulting in a total of sixteen tests. This was also important to see the impact of these parameters on the efficiency of our approach. Typically, we want to know if there are better settings than others, so if there are settings that generate smaller result sets or are there types of settings that need be avoided. Due to a lack of space we discuss in the following only the most interesting ones. The complete result set and the description of the different test settings are available in Appendix B.

TABLE 6.1 – Different settings

	Interval		**Weighting**				**Thresholds**			
	Min	Max	P1	P2	P3	P4	T1	T2	T3	T4
Cost	10	1000	0.40	0.40	0.85	0.25	100	100	100	50
Risk	1	5	0.20	0.20	0.05	0.25	1	1	2	1
QoS	1	10	0.25	0.30	0.05	0.25	2	1	4	4
Complexity	0	1	0.15	0.10	0.05	0.25	0.2	0.1	0.4	0.1

One first point to notice is that the type of distribution directly impacts the calculation time needed for generating the final solution set (as shown in FIGURE 6.6). A normal distribution allows to exclude more rapidly configurations from the final result set than with a uniform distribution. The overall average calculation time is of 0.893 seconds for a uniform distribution, whereas it is of 0.784 seconds for a normal distribution. This can be explained by the fact that a normal distribution generates configurations with characteristics that a globally closer one to another. Therefore, the threshold conditions are more often met than for a uniform distribution. In general, in a uniform distribution, it is more probable to get a configuration that has one criterion with a value that exceeds those of the other configurations. Thus, while the other conditions of the hybrid-domination are met, the condition for this criterion is not, and the configuration is not dominated (and cannot be excluded directly).

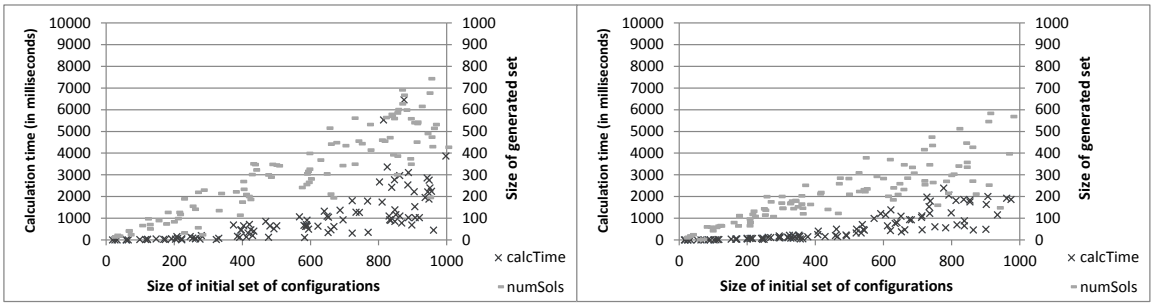


FIGURE 6.6 – Calculation time (*calcTime*) and number of generated solutions (*numSols*) with setting P1-T1 for both distributions: uniform (left), normal (right)

Additionally, the same observation can be made for the exclusion ratio. Globally, for a normal distribution more configurations are excluded than for a uniform distribution (as shown in FIGURE 6.7). Thus, the result sets are smaller. The overall average exclusion ratio is of 43.43% for a uniform distribution, whereas it is of 53.84% for a normal distribution. This observation can be explained by the same behaviour explained previously. In a uniform distribution, the characteristics of the confi-

gurations are closer one to another. Therefore, the threshold conditions are more often fulfilled.

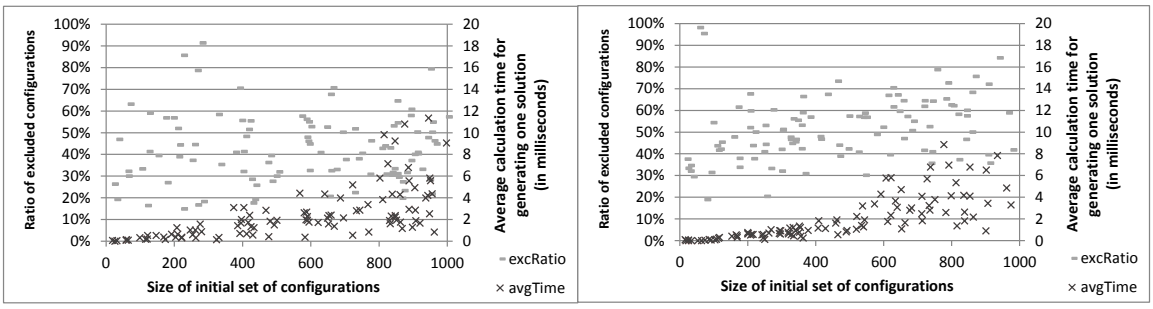


FIGURE 6.7 – Exclusion ratio (*excRatio*) and average calculation time for generating one solution (*avgTime*) with setting P1-T1 for both distributions: uniform (left), normal (right)

However, FIGURE 6.7 also shows that there is no correlation between the size of the initial set of available configurations and the number of excluded configurations. For all tests, the exclusion ratio is similarly distributed regardless the size of the initial set. Therefore, no predictions can be made regarding the number of configurations that will be excluded when considering the size of the initial set.

Concerning the weightings and the thresholds, they influence directly both measured values: the average exclusion ratio (see FIGURE 6.8) and the average calculation time per configuration (see FIGURE 6.9).

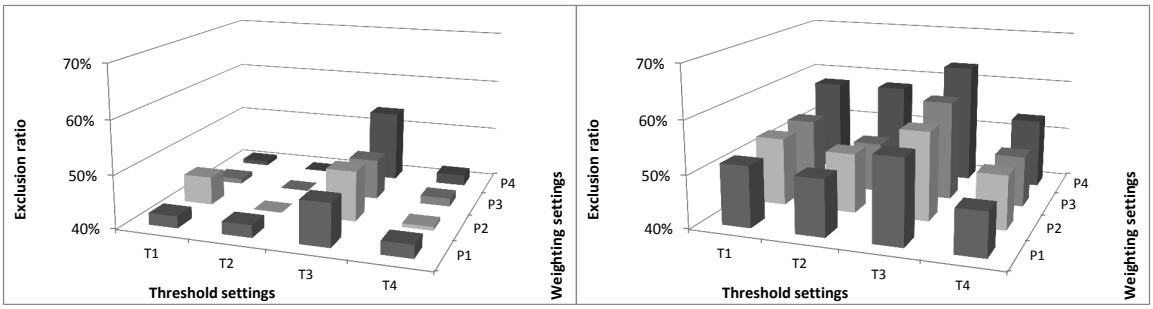


FIGURE 6.8 – Average exclusion ratio for both distributions: uniform (left), normal (right)

We can notice that the most interesting results are generated with the T3-threshold setting. With this setting we achieve the best average exclusion ratio and the lowest average calculation time. This can be explained with the types of thresholds given as input. Typically, T3 has the highest threshold values, which means that it becomes more difficult for a configuration to contradict the weighting-domination. With lower thresholds it becomes easier for one criterion to have such a good value that it is above the specified threshold. We deduce from this observation that specifying high threshold values will enhance the performance of our algorithm.

Another interesting observation is that the P4-weighting setting is interesting for a normal distribution. Indeed, it has globally a higher exclusion ratio and a lower calculation time than the other weighting settings. Typically, P4 has an equal weighting value for all criteria, which means that all criteria are taken equally into account. With four criteria, this means that in order to have a domination between two configurations, at least three values must be better (no matter which ones). Leaving only one criterion that can contradict the weighting-domination. Thus, a domination is more easily achieved (because it is more difficult to create a contradiction with the thresholds), and configurations can be excluded more rapidly.

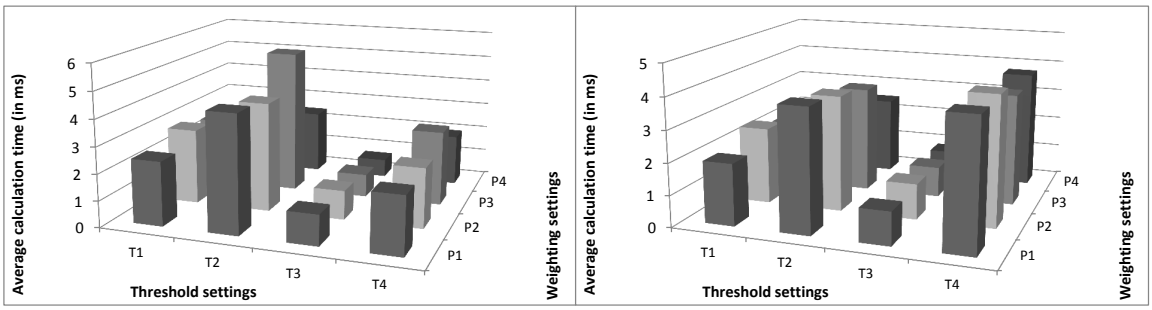


FIGURE 6.9 – Average calculation time for both distributions: uniform (left), normal (right)

Finally, we can notice the the P3-weighting setting is the least interesting one for a uniform distribution. It has globally the lowest exclusion ratio and the highest calculation time. The difference with the P3 setting is that the focus is put on one single criteria (the costs). Therefore, in opposition the the other settings, there are more criteria that can contradict the weighting-domination. Thus, it becomes more difficult to achieve a domination at all, and this lengthens the duration of the algorithm.

6.1.2.3 Discussion

The objective of our optimization algorithm was to compare different possible configurations while considering multiple criteria. We also stated that there was no need for generating only one single solution for a given set, but that the existence of a broker-consumer relationship motivates the idea of generating a set of possible solutions. Therefore, our algorithm tries to exclude solutions that can be considered as non-optimized regarding their criteria. In this sense, our algorithm behaves as expected.

Our multi-criteria algorithm allows to integrate effectively the risk with the other criteria to compare the possible deployment configurations. In opposition to other approaches, like the linear scalarization method, there are no complex aggregation formulas to define. This is especially interesting because the approach becomes generic, and needs very few adaptations for different use cases (the local thresholds may need an adaptation to comply with different scales). Moreover, all criteria can be considered as optimization criteria, since they are not transformed into constraints like in the ϵ-constraint method.

One could argue that the number of excluded solutions for the different runs seems to be insufficient. An overall exclusion ratio of 43.43% for a uniform distribution cannot be considered as being high enough. Knowing that the initial goal was to help the cloud broker to select the best configuration among a large set of possibilities, our algorithm does not really make the problem less complex. It does not even reduce the set by half of its size. However, the values are randomly generated and do not correspond to any real world use case. Indeed, when considering the normal distribution that simulates a real world use case a bit more accurately, the overall exclusion ratio increases to 53.84%.

Additionally, we suggest that it could make sense to further investigate this approach to propose some improvements. For example, we did not test the behaviour of the algorithm when considering more criteria than only four. Additional observations could also be made when changing the intervals for generating the random values. And most importantly, the approach should be tested against real values.

6.1.3 Process deployment

As already explained in the previous sections, our tool is part of a more global approach for decomposing business processes into fragments. Existing work shows precisely how such a fragmentation is done, and tools that implement these approaches already exist [FYG09]. Therefore, we do not show the parts of the developed tool that correspond to the fragmentation of a business process, since it cannot be considered as a scientific contribution of this thesis. However, one aspect specific to our cloud context has been developed in the frame of this thesis: the automatic deployment of fragmented processes to multiple cloud offers.

6.1.3.1 Presentation of the tool

Since it can be very time-consuming to deploy fragments of business processes on multiple locations (especially when there are many fragments), we developed an interface supporting the automatic deployment and monitoring of fragmented business processes. This tool can be linked to remote cloud providers using web services to work with different offers that propose a business process execution platform. The tool has been developed in JAVA.

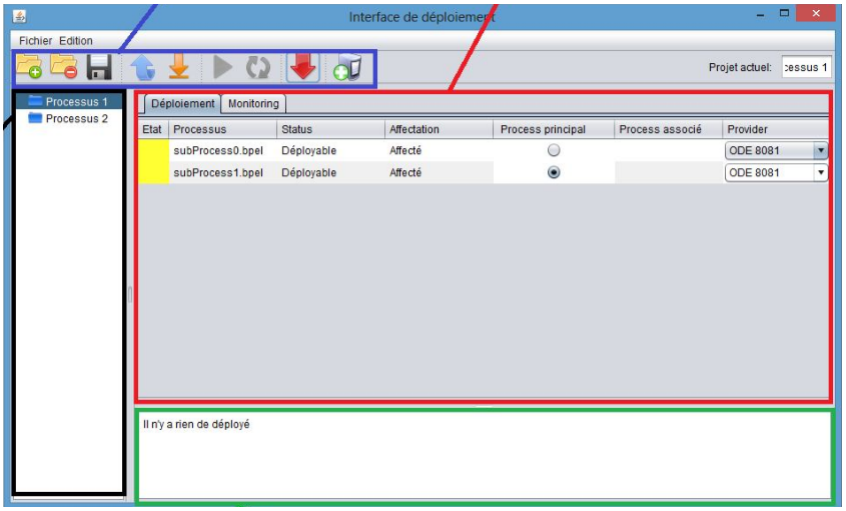


FIGURE 6.10 – Interface of the deployment and monitoring tool

The interface (shown in FIGURE 6.10) is divided in four main parts: the **tool bar**, the **process list**, the **console** and finally the **deployment and monitoring area**.

The tool takes as input fragmented business processes that are encoded in BPEL. Each fragment is defined with two files: the WSDL file that describes the web service interfaces of the process, and the core BPEL file that describes the logic of the process. Therefore, the entire process is composed of multiple pairs of such files that can be combined in a common archive file. Such an archive file can be used as input for tool, each independent process appears on the left in the **process list**.

Once such a process opened, the different fragments appear on the right (in the **deployment and monitoring area**). Under the **deployment** tab, each fragment can be assigned to a cloud provider through a drop-down list. This interface allows also to define the *initiator process* that will allow to initiate a process instance. These changes are directly propagated to the BPEL and WSDL files that have to change accordingly. Once every fragment affected to a cloud offer, the entire process can be deployed using the **Deploy** button in the upper **tool bar**.

The **monitoring** tab allows to see the details of the executed instances: their status (completed, active, stopped, *etc.*), the time of execution of each activity and the content of the sent and received

messages. New instances can also directly be started with the **start** button of the upper **tool bar**. This is especially useful for a testing purpose.

6.1.3.2 Case study deployment in the cloud and experimentation report

Moreover, we conducted an experiment to demonstrate the proper functioning of the tool and the feasibility of an execution across multiple environments. In our experiment we selected two remote clouds and one local cloud that execute their proper business process execution engines. We used a process fragmented in three partitions to simulate the different kinds of clouds regarding security properties. The deployment of our partitioned business process is as follows:

- the first sub-process is hosted on a public version of the cloud platform: like simple web services, services are accessible through HTTP on a public URI,
- the second sub-process is hosted on a secured version of the cloud platform: services are only accessible authenticated through a SSL connection,
- the third sub-process is hosted on a private cloud: we have installed an instance of the business process execution engine on a local architecture to simulate a private cloud, entirely controlled by us.

However, while the deployment of BPEL process models has gone well, we encountered some problems with message synchronization. If the remote cloud platform includes a messages store to queue incoming messages and relays them at an appropriate time, as requested for sub-processes synchronization, we experimented an unexpected behaviour: some messages were lost and the message ordering was not always enforced. In other words, event causality was not enforced.

A classical way to surpass this limit is to implement/reuse message queues associated to each process and achieving exactly-once in-order message reception, as example as defined by the WS-ReliableMessaging [OAS09] standard. Due to lack of time, but to nevertheless validate the feasibility of the approach, we intervened directly in the generated code to manage synchronization by hand. However, we argue that it does not prevent the usefulness of our approach, as this limitation is specific to the platform used for performing our tests, and that such issues should be fixed in a very near future.

6.2 Use case in access control

To see the limits of our model and take some first steps in the automated consideration of security risks in information systems, we implemented our cloud security risk assessment model in a access control use case [BGPG15]. The idea is to enhance existing access control systems with additional risk metrics that could make such systems more secure.

6.2.1 Context

In parallel to our work, an industrial project tool place at the LORIA [57] with a company developing a Professional Social Network (PSN) for cloud environments. One objective of this project was to study access control mechanisms with security policies defined in a distributed fashion. We took this opportunity to propose further enhancements by considering the security risk of such contexts.

57. Laboratoire lorrain de recherche en informatique et ses applications: `http://www.loria.fr`

More specifically, PSNs are collaborative platforms that offer several ways to companies to collaborate, like exchanging information, working on common projects through shared tools, organizing meetings, *etc.*. These collaborations can be intra- or inter-organizational (so also include multiple companies). A detailed description of such environments is given in [Row14].

One benefit of PSNs is that users become autonomous regarding there work, they are able to share their resources with other users through the platform. The user defines himself with whom he wants to share his data and with whom not. Basically, these rules correspond to access control policies: a *subject* has the right of performing an *action* on a specific *resource*. With such an autonomy of the user, access control policies can become very fine-grained and well adapted to the user's needs. However, the company cannot entirely confide to its users the definition of trusted policies. Malicious insiders or insufficiently informed employees can pervert such systems by defining faulty policies. The need of frequent changes of the policies in such environments makes it not an option for the company itself to define the policies. Policies defined at the level of the organization would be too static and not fine-grained enough. These contradictory requirements of flexibility (fine-grained policies) and control (organizational policies defined with a high level of abstraction) makes the design of new architectures necessary.

6.2.2 Formal framework

To deal with these challenges, we take a risk-based approach to provide to the company means to control there resources while keeping the flexibility of PSN environments. Typically, the risk to prevent can be defined as follows: *a user gets unauthorized access to a resource of the system*. Different reasons could lead to such an event:

- the initial access control policies have been erroneously defined
- an attacker impersonates a trustworthy user (or an existing user becomes malicious) and tries to steal valuable information
- the sensitivity of a resource changes over time, making the existing access control policy outdated
- the authentication mechanism includes security flaws, so the user's identity cannot be guaranteed

With a risk perspective in accordance of our cloud security risk assessment model, these threats can typically be evaluated in the form of their **impact**, the **vulnerabilities** that facilitate their occurrence and the overall **probability** that they occur. However, in opposition to the model presented in Chapter 4, here we need to assess the risk at run-time, so dynamically and not before the deployment of the system. Indeed, we propose to evaluate for each incoming request its "riskiness" and reject it if its too high. Thus, we redefine partially some concepts to properly assess the risk of the incoming requests.

6.2.2.1 Impact

The impact of an unauthorized access depends completely on the resource that is being requested. Some resources are more important to the company than others. Moreover, some actions on a resource can have more serious consequences than others. A high confidential information should not be easily readable, however if it can be easily re-generated, an accidental deletion is not a big issue. Therefore, we formalize the impact as in Formula 6.4.

Definition 21 (Impact) *Depends on the action being requested on a given resource. More the resource is important more the impact will be high. An important resource is a resource on which little access permissions are defined. Accordingly, for a given request req that implies user u, resource r and action a, we compute the average of the policy responses for requesting action a on the resource r regarding to all the users of the company. For this purpose, we check how much permission(s) are given to perform the action a on the resource r within the company:*

$$Impact(a,r) = 1 - \frac{\sum_{u \in User} Policy(u,a,r)}{Card(User)} \tag{6.4}$$

while

$User$ *is the set of users of the company,*

$a \in Action$, *the set of actions (i.e. {R, W, X}),*

$r \in Resource$, *the set of resources of the company,*

$Policy$, *the policy-decision ($accept$=1, $reject$=0),*

$Card(User)$, *the number of users of the company.*

6.2.2.2 Vulnerability

The vulnerabilities that may lead to an unauthorized access are generally due to a flaw in an authentication mechanism. To access a PSN environments, each company can implement its own mechanism. Basically, some authentication mechanisms are more secure than others: a two step identification is more difficult to trick than a simple password login. Thus, on the same platform there can be users with different guarantees regarding their identify. The collaborative platform itself can also include some vulnerabilities, but for the sake of simplicity we limit the formalization to the authentication mechanism to that of Formula 6.5.

Definition 22 (Vulnerability) *Depends on the strength of the authentication mechanisms used to identify a user. E.g., $Auth$ = {Guest, PIN, Login/password, OAuth, 2 Step Validation, Biometric}. Each mechanism gets a score that represents the strength level of the authentication mechanism. Thus for a given request req*

$$Vulnerability(req) = V(C) = Score(A_C) \tag{6.5}$$

while

C, *the company of the user making the request req,*

A_C, *the authentication mechanism of C,*

$Score : Auth \rightarrow [0,1]$, *the strength level of A_C*

6.2.2.3 Threat

The threat emanates from the user, since it is his action (the acceptance of the request) that can generate a negatively affect the company. To evaluate the probability of an attack (or an unintentional adverse action), we evaluate the trustworthiness of a user. *E.g.*, a user who often try to perform unauthorized actions should not be trusted as much as a user who has an exemplary behaviour.

A suspicious request from the second user is less risky than one from the first user. A reliable trust computation is out of scope of this thesis, but existing work [Zac99] propose approaches in directions as defined in Formula 6.6.

Definition 23 (Threat) *Its probability depends on the trustworthiness of the user making the request. More the user u is trusted, less the threat will be probable. As the trust values are belonging to the interval* $]0,1[$ *, we interpret this by the formula:*

$$Threat(u) = 1 - Trust(u) \tag{6.6}$$

6.2.2.4 Risk

These three types of information can than be combined to evaluate the risk of an incoming request. Thus, this risk value integrates three different concepts: the probability that the user is up to something bad (intentionally or not), the impact that the action could have if it is not genuine and the vulnerabilities that could lead to such an event. This is formalized in Formula 6.7.

Definition 24 (Risk) *In our context we decide to take a linear approach for calculating the risk value. However, in some contexts, it may be interesting to focus more on one value than on another. Thus, we introduce weightings into the risk-formula:*

$$Risk(req) = \frac{k_v \times V(C) + k_t \times T(u) + k_i \times I(a,r)}{k_v + k_t + k_i} \tag{6.7}$$

while

V*, the Vulnerability, and* k_v *its weighting,*

C*, the company of the user making the request* req*,*

T*, the Threat, and* k_t *its weighting,*

u*, the user of the request* req*,*

I*, the Impact, and* k_i *its weighting,*

a*, the action of the request* req*,*

Basically the weightings can all be set to 1 to take equally into account all values (thus the risk corresponds to the average). However, some organization could put their focus more on the impact.

Organizations still keep the control of the access to their resources (shared by users) by determining the maximum risk level they accept. Each organization is free to set (manually) a threshold value beyond which the incoming requests will be rejected.

6.2.3 Results

To demonstrate the feasibility of the approach and also test its usefulness we conducted different experiments. The details of the results are not presented in this work, since it is not in the scope of this thesis. A detailed description of the evaluations and the results are published in [BGPG15]. However, we draw the following conclusions of the implementation:

- the risk adapts to the system behaviour and the access control policies defined by the users of the company. Typically, when there are many rejections there are more reasons to be cau-

tious. In this sense, the approach rejects more requests in such type of systems (for the same threshold).

- the global influence of the introduction of the risk metrics seems fair (as long as the threshold remains reasonable). Thus, there is not a huge amount of rejected requests that is introduced to the system.
- the risk metrics is globally coherent to the access control policies. Request that are rejected based on the risk are fairly overlapping with the requests that are already rejected based on the policy.

These experimentations show that the cloud security risk assessment model is adaptable to other types of context than simply comparing cloud providers. The systematic differentiation between the impact, the threat and the vulnerabilities, even in systems with different entities (with contradictory objectives) is coherent. Moreover, it gives good perspectives to extend our approach to a dynamic context and to an automated assessment of the impact.

6.3 Case study

During the thesis project we found a company confronted to the problem of migrating their IT infrastructure to cloud services. Rapidly the company had to face the problem of comparing different cloud offers regarding their security. After discussion with the different partners, we found that this was a good case study to validate our cloud security risk assessment model. Therefore, we detailed our approach to the company in order to apply the cloud security risk assessment as defined in Chapter 4. The first goal was obviously to help the company to make their final decision, but also to validate (at least partially) the work developed in this thesis.

6.3.1 Overview

The project of the case study took place in a lawyer's office in Belgium with about thirty employees. The company has a classic IT infrastructure on-premises managed by an external and trusted IT consulting company. Such type of infrastructure has been identified as being no longer adapted to the needs of the company, especially regarding the security requirements and the mobility of the employees. Therefore, the company decided to analyse the possibility of migrating their infrastructure to cloud services. However, the knowledge and technical expertise for carrying out such an analysis is clearly not in the competencies of the company. Therefore, the lawyer's office contacted their IT consulting firm to realize this study.

In this objective, the consultancy company launched a call for projects to find possible cloud offers that could answer the lawyer's office's requirements. Different providers answered this call, and the consultancy company had to assess their security level to make a proper proposition. As the situation presents, and to link the study to our model, the lawyer's office takes the role of the cloud consumer, whereas the consultancy firm plays the role of the cloud broker.

6.3.2 Cloud security assessment

Only a few of the responding providers are retained, based on several criteria not discussed in detail here. But typically, criteria like reputation, expertise or even motivation to answer those questionnaire are taken into account for pre-selecting the providers that are worth considering. In the following, we limit the study to two providers that are compared using the CAI-Questionnaire

defined by the CSA (see Appendix A). The first thing to note is that the two providers were willing to answer the requests for filling out these questionnaires and that no additional motives were required. In the following, and for anonymity reasons, these two providers will be denoted as *Provider A* and *Provider B*.

By analysing the answers to these questionnaires, we are able to compare the two solutions based on the CSA Cloud Control Matrix [CSA14]. Overall, *Provider A* implemented 171 of the 295 security controls specified by the CSA, and *Provider B* implemented 180 of them. Clearly, a difference of 9 controls among a set of 295 is insignificant. Therefore, the study was refined according to the model defined in Chapter 4. Each control was related to one of the 9 CSA cloud threats as advised in [CSA13]. The relation between the threats and the security controls is given in TABLE 6.2.

TABLE 6.2 – List of security controls mitigating the 9 CSA cloud threats

Threats	**List of mitigating controls**
Data Breaches	BCR-11, DS1-07, GRM-02, EKM-02, EKM-03, IAM-14, AIS-04, IVS-08
Data Loss	BCR-11, GRM-02, BCR-06
Account or Service Trafic Hijacking	IAM-02, IAM-09, IAM-11, IAM-10, IAM-12, IVS-01
Insecure Interfaces and APIs	IAM-09, AIS-04, AIS-01
Denial of Services	GRM-01, IVS-04, BCR-08, AIS-01
Malicious Insider	DSI-04, DSI-06, DCS-09, DCS-08, DCS-04, DCS-04, GRM-07, IAM-09, IAM-10, HRS-07, IAM-05, EKM-02, EKM-03, IVS-09, GRM-11
Abuse of Cloud Services	SEF-04, HRS-08
Insufficient Due Diligence	GRM-02, GRM-11, GRM-10, AIS-04, AIS-01, IVS-13, IVS-06, IVS-09, IVS-04, GRM-01, IAM-08, BCR-09, BCR-01
Shared Technology Vulnerability	DSI-04, IAM-12, IVS-09, IVS-01, GRM-01, IAM-02, IAM-05, EKM-03, TVM02

Now that these mitigations are defined, the two providers can be compared more precisely, based on the 9 threats. The coverage is brought on a score of 10 possible points: if all mitigating controls are implemented, the score is equal to 10, none 0, half of the required controls 5, and so on. Results of this comparison are shown in FIGURE 6.11.

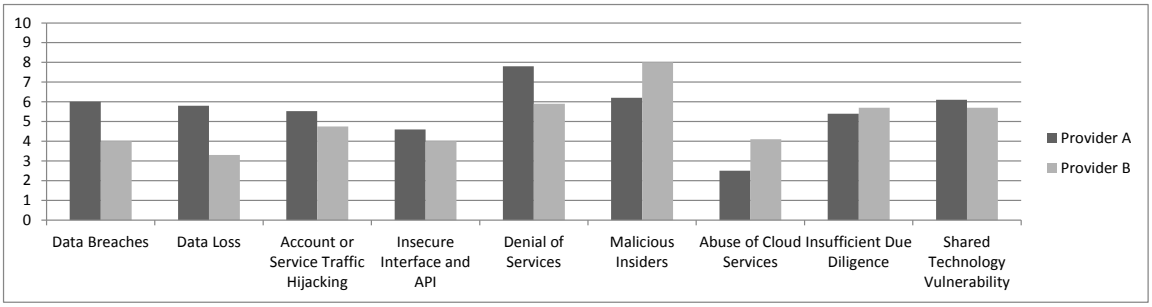


FIGURE 6.11 – Coverage level of the two providers

We can observe that the difference between the two providers is more significant than previously assumed. The average coverage score of *Provider A* is 5.55 whereas it is of 5.05 for *Provider B*. Moreover, the two providers focus on different areas regarding security (*e.g.*, when comparing the *Denials of Service* threat with the *Malicious Insider* threat).

According to the model presented in Chapter 4, these values could be more refined when adding the threat's probability. However, these information only slightly change the final values for the different threats, and as such precise values are not needed, this additional weighting is disregarded.

Furthermore, a multi-cloud environment is not considered in this project. Thus, it becomes not necessary to identify in detail the different assets of the company. One *primary asset* is defined that encapsulates the security needs of all data and processes that would be moved to the cloud. The need of this asset is evaluated on the three CIA (Confidentiality, Integrity, Availability) security criteria with three levels: $\{Low = 0.0, Medium = 0.5, High = 1.0\}$. The need of this *primary asset* is given in TABLE 6.3.

The consequences of the nine threats are specified through simple binary relations. So basically, does the threat affect the criterion (= 1) or not (= 0). The relations strictly follow the indications given by the CSA in [CSA13] and are given in in TABLE 6.3.

TABLE 6.3 – List of security controls mitigating the 9 CSA cloud threats

	Confidentiality	**Integrity**	**Availability**	**Harm**
Security need				
Primary asset	High	High	Medium	
Consequences				
Data Breaches	Yes			1.00
Data Loss			Yes	0.50
Account or Service Trafic Hijacking	Yes	Yes	Yes	1.00
Insecure Interfaces and APIs	Yes	Yes		1.00
Denial of Services			Yes	0.50
Malicious Insider	Yes	Yes	Yes	1.00
Abuse of Cloud Services	Yes	Yes	Yes	1.00
Insufficient Due Diligence	Yes	Yes	Yes	1.00
Shared Technology Vulnerability	Yes	Yes	Yes	1.00

With these additional information, the comparison can be even more precise, since it allows to evaluate the providers according to the needs of the lawyer's office (and not regarding their security in general). Therefore, it allows to decide which of both offers would be more adapted to migrate to (see FIGURE 6.12). Indeed, the *Denial of Service* threat is now significantly reduced when considering the formula defined in Chapter 4. In this perspective, there is a non-negligible difference regarding the cloud threats, because the *Malicious Insider* threat on *Provider B* is too high for being accepted as it is.

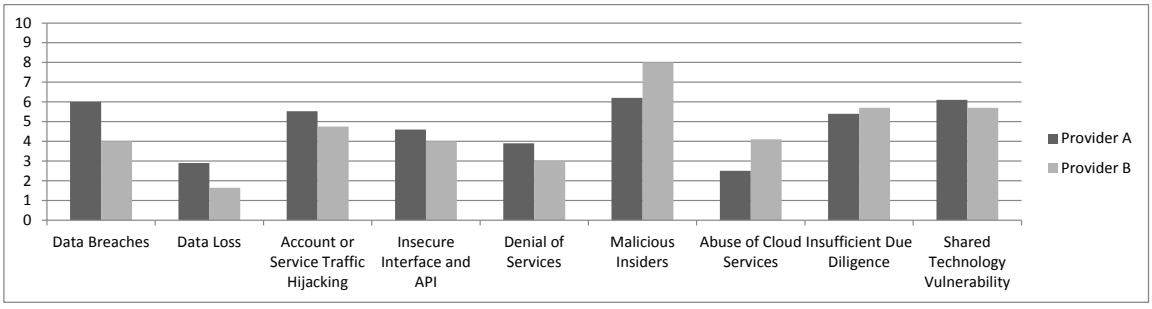


FIGURE 6.12 – Risk levels of the two providers

6.3.3 Results and discussion

Regarding the results obtained on this case study, the consultancy company advises a migration to the cloud of *Provider A*. As defined in our global approach, the broker's role is not to make decisions, but more to advise the cloud consumer based on his expertise. The final decision is taken by the cloud consumer (*i.e.* the lawyer's office) and it was still not fixed at the time of writing of the manuscript. But first discussions gave the impression that this advise would be respected.

Globally, the feedback of this study was very positive, since the assessment model seemed to be very adapted to such type of comparisons. Obviously (and as already stated previously), the comparison took place on many other criteria (*e.g.*, cost, reputation of the provider, complexity of the migration, *etc.*) to make a pre-selection. Since it is not the scope of our work, providers that were excluded for other reasons were not presented here. However, this assessment was an interesting asset for the comparison, the major benefit being that it allowed to formalize the security aspects of the final decision. In general, it is more the result of guesswork when taking into account security for selecting such type of offers. Here the main advantage is to put metrics on security and to be able to compare different offers in a formal way, at least regarding the security risk.

Moreover, the IT consultancy company notified their interest in working on the development of a tool helping in the automating of such assessments. Further improvements regarding a more detailed treatment of the answers to the questionnaires are also considered.

Chapter 7

Conclusion

7.1 Review of the contributions

In this thesis we made different contributions to support the trusted deployment of business processes in cloud environments. First, a method supports the semi-automatic adaptation of process models based on security objectives and organizational constraints to transform existing business processes into risk-aware processes for cloud environments (see Chapter 3). Second, a cloud security risk assessment model supports the evaluation of risk when deploying applications into cloud environments. It takes into account annotations on process models to evaluate their security need, and information coming from the providers to assess their response to given security risks (see Chapter 4). Finally, we presented a framework that supports the automated deployment of business processes to multiple cloud environments. We included in this existing work a multi-criteria optimization algorithm to select the configuration that fits well different quality of service criteria (see Chapter 5).

Overall, we claim that our contributions help to increase the trust that companies can have when deploying their processes into the cloud. The global strategy of our approach is twofold. On the one hand we adapt the processes to better handle the security threats related to cloud computing. On the other hand we assess the security level of cloud providers to quantify the risk and select the best possible configuration. This approach is supported by a tool to automate the transformation, the assessment and the deployment that takes into account other important criteria when considering a cloud outsourcing.

The validation of our work has been conducted in different ways. First, to demonstrate the feasibility of our approach, we implemented it on different prototypes: an interface for evaluating cloud providers regarding their security, a multi-criteria optimization algorithm integrated in a business process decomposition framework and a tool for automatically deploying and monitoring business processes in cloud environments (see Section 6.1). Second, we adapted our security risk assessment model to enhance existing access control systems with risk metrics to make them more flexible, dynamic and context-aware (see Section 6.2). Third, we conducted a comparison of real cloud providers with our security risk assessment model to advise an existing company on a possible migration (see Section 6.3). Additionally, the work has been published in different peer-reviewed scientific articles and has been presented to international conferences.

7.2 Limitations and perspectives

To address some limitations of our work, we propose different perspectives that could improve or deepen the work proposed in this thesis.

Run-time changes One of the first aspects that has not been addressed in our work is the dynamic nature of cloud computing. Our approach takes a perspective *at design-time* and does not natively integrate the change of either the environment or the process itself. Indeed, not only can the available cloud offers evolve (in terms of costs, quality of service but also security), even the security risks themselves can change. Therefore, the selected configuration may not be as optimized as it seemed to be at the time of the deployment. It would be interesting to take such changes into account, like dynamic re-deployment of a process or even run-time deployment (on-the-fly). First steps in this direction were taken by the Cloud Security Alliance that proposes different certification levels. One of them certifies the continuous monitoring of the security level of the cloud provider. In such a context, changes could be automatically detected through an event-driven approach. The work presented in Section 6.2 shows that the cloud security risk assessment model is appropriate for working in a dynamic context. Moreover we argue that the design-decision of using heuristics for selecting an optimized configuration (see Chapter 5) is also motivated by this perspective: the time needed for finding a good deployment solution is an important factor when working in a run-time context.

Automatic definition of security needs Another aspect not explored in this thesis is the precise definition of security needs. We followed the guidelines of the most important security risk assessment methods by assuming that these are given manually by a security expert. However, these information may already exist somewhere else, and it would be interesting to generate them automatically. Different formal models for defining security requirements exist (like Secure Tropos [MMGG03]) that could be used to explore these perspectives. Indeed, one major drawback of our current approach is that these security needs are subjective: two different persons could define different needs. By generating them automatically, the security levels would no longer be open to interpretation. Or at least, prevent to carry out the same task multiple times (since security information are probably given at some other point when defining the process).

Obfuscation The last, and probably the most interesting perspective is that of obfuscating process models before deploying them in cloud environments. A technique already used in programming to hide the source code of applications, could be a source of inspiration for hiding the mechanics of a business process. For that purpose, business process models need to be automatically evaluated regarding their complexity. This could lead to the identification of the critical areas of the process that should be protected the most. Also, security patterns, like redundancy, false tasks/messages or separation constraints could be developed. Such patterns could then be automatically integrated into existing processes to increase their complexity and reduce the probability for an attacker to understand the business process. First advances in this direction have already been published in [GANYG15].

This contribution proposes to analyse business processes regarding their control flow and especially the type of gateways that define its execution behaviour. The paper proposes to classify existing gateways in two distinct categories: the *decisions* and the *synthesis*. This allows to define a coarse-grained identification of critical blocks in business processes and thus propose fragmentation rules to hide certain relationships between tasks or even paths. The different types and relationships are formalized to generate the fragmentation rules in a semi-automated way. The overall objective is to preserve the know-how of a business process before deploying it in a cloud environment.

Bibliography

[AAHR10] Wil M.P. Van Der Aalst, Michael Adams, Arthur Ter Hofstede, and Nick Russell. *Modern Business Process Automation - YAWL and its Support Environment*. Springer Berlin Heidelberg, 2010.

[Aal11] Wil M.P. Van Der Aalst. Business process configuration in the cloud: How to support and analyze multi-tenant processes ? In *Web Services (ECOWS), 2011 Ninth IEEE European Conference on*, pages 3–10, Sept 2011.

[AFG+09] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy H. Katz, Andrew Konwinski, Gunho Lee, David A. Patterson, Ariel Rabkin, and Matei Zaharia. Above the clouds: A berkeley view of cloud computing. Technical report, 2009.

[AJG+15] Abdelzahir Abdelmaboud, Dayang N.A. Jawawi, Imran Ghani, Abubakar Elsafi, and Barbara Kitchenham. Quality of service approaches in cloud computing: A systematic mapping study. *Journal of Systems and Software*, 101:159–179, mar 2015.

[ALMS09] T. Anstett, F. Leymann, R. Mietzner, and S. Strauch. Towards bpel in the cloud: Exploiting different delivery models for the execution of business processes. In *Services - I, 2009 World Conference on*, pages 670–677, July 2009.

[AM13] Naved Ahmed and Raimundas Matulevicius. A taxonomy for assessing security in business process modelling. In *RCIS*, pages 1–10, 2013.

[AM14] Naved Ahmed and Raimundas Matulevičius. Securing business processes using security risk-oriented patterns. *Comput. Stand. Interfaces*, 36(4):723–733, Jun 2014.

[AZ/04] AS/NZS 4360 SET Risk Management, Australian/New Zealand Standards, 2004.

[BBDA12] Mehdi Bentounsi, Salima Benbernou, Cheikh S. Deme, and Mikhail J. Atallah. Anonyfrag: an anonymization-based approach for privacy-preserving bpaas. In *1st International Workshop on Cloud Intelligence (colocated with VLDB 2012), Cloud-I '12, Istanbul, Turkey, August 31, 2012*, page 9, 2012.

[BePP98] Rainer E. Burkard, Eranda Çela, Panos M. Pardalos, and Leonidas S. Pitsoulis. The quadratic assignment problem. In *Handbook of Combinatorial Optimization*, pages 241–238. Kluwer Academic Publishers, Dordrecht, 1998.

[BGPG15] Ahmed Bouchami, Elio Goettelmann, Olivier Perrin, and Claude Godart. Enhancing access control with risk metrics in collaborative federated cloud environments. In *14th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, TrustCom 2015, Helsinki, Finland, August 20-22, 2015*, 2015.

[BH14] Amid Khatibi Bardsiri and Seyyed Mohsen Hashemi. Qos metrics for cloud computing services evaluation. *International Journal of Intelligent Systems and Applications (IJISA)*, 2014.

[BK13] Niyati Baliyan and Sandeep Kumar. Quality assessment of software as a service on cloud using fuzzy logic. In *2013 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM)*. IEEE, oct 2013.

[BKNT11] Christian Baun, Marcel Kunze, Jens Nimis, and Stefan Tai. *Cloud Computing: Web-Based Dynamic IT Services*. Springer Publishing Company, Incorporated, 1st edition, 2011.

[BL74] D. Bell and L. Lapadula. Secure computer systems: mathematical foundations and model. 1974.

[BLB15] Matthias Becker, Sebastian Lehrig, and Steffen Becker. Systematically deriving quality metrics for cloud computing systems. In *Proceedings of the 6th ACM/SPEC International Conference on Performance Engineering - ICPE'15*. ACM Press, 2015.

[BYOG13] Kahina Bessai, Samir Youcef, Ammar Oulamara, and Claude Godart. Bi-criteria strategies for business processes scheduling in cloud environments with fairness metrics. In *IEEE 7th International Conference on Research Challenges in Information Science, RCIS 2013, Paris, France, May 29-31, 2013*, pages 1–10, 2013.

[CAM10] Common Assurance Maturity Model Guiding Principles. `http://www.common-assurance.com/resources/Common-Assurance-Maturity-Model-vision.pdf`, 2010.

[CBT11] Shankar Babu Chebrolu, Vinay Bansal, and Pankaj Telang. Top 10 cloud risks that will keep you awake at night. Technical report, 2011. `https://www.owasp.org/images/4/47/Cloud-Top10-Security-Risks.pdf`.

[CFBH07] B. Carminati, E. Ferrari, R. Bishop, and P.C.K. Hung. Security conscious web service composition with semantic web support. In *Data Engineering Workshop, 2007 IEEE 23rd International Conference on*, pages 695–704, April 2007.

[Clo14] Cloud Security Alliance. Security, Trust and Assurance Registry. `https://cloudsecurityalliance.org/star/`, 2014.

[CLRA13] Raffaele Conforti, Massimiliano De Leoni, Marcello La Rosa, and Wil M.P. Van Der Aalst. Supporting risk-informed decisions during business process execution. In Camille Salinesi, MoiraC. Norrie, and Óscar Pastor, editors, *Advanced Information Systems Engineering*, volume 7908 of *Lecture Notes in Computer Science*, pages 116–132. Springer Berlin Heidelberg, 2013.

[CRDRB09] Rodrigo N Calheiros, Rajiv Ranjan, César AF De Rose, and Rajkumar Buyya. Cloudsim: A novel framework for modeling and simulation of cloud computing infrastructures and services. *arXiv preprint arXiv:0903.2525*, 2009.

[CSA13] The Notorious Nine - Cloud Computing Top Threats in 2013. Technical report, Cloud Security Alliance, 2013. `https://downloads.cloudsecurityalliance.org/initiatives/top_threats/The_Notorious_Nine_Cloud_Computing_Top_Threats_in_2013.pdf`.

[CSA14] Cloud Control Matrix. Technical report, Cloud Security Alliance, 2014. https://cloudsecurityalliance.org/download/cloud-controls-matrix-v3/.

[DRA05] Alexander Dreiling, Michael Rosemann, and Wil M.P. Van Der Aalst. From conceptual process models to running workflows : a holistic approach for the configuration of enterprise systems. In *PACIS'05*, pages 363–376, 2005.

[DRMR13] Marlon Dumas, Marcello La Rosa, Jan Mendling, and Hajo A. Reijers. *Fundamentals of Business Process Management*. Springer, 2013.

[EJF+14] Seven Euting, Christian Janiesch, Robin Fischer, Stefan Tai, and Ingo Weber. Scalable business process execution in the cloud. In *2014 IEEE International Conference on Cloud Engineering, Boston, MA, USA, March 11-14, 2014*, pages 175–184, 2014.

[ENI09a] Benefits, risks and recommendations for information security. Technical report, European Network and Information Security Agency, 2009. http://www.enisa.europa.eu/activities/risk-management/files/deliverables/cloud-computing-risk-assessment/at_download/fullReport.

[ENI09b] Information Assurance Framework. Technical report, European Network and Information Security Agency, 2009. https://www.enisa.europa.eu/activities/risk-management/files/deliverables/cloud-computing-information-assurance-framework/.

[Eur12] EuroCloud Deutschland eco e.V. Eurocloud Star Audit. https://eurocloud-staraudit.eu/, 2012.

[FDGG14] Walid Fdhila, Marlon Dumas, Claude Godart, and Luciano García-Bañuelos. Heuristics for composite web service decentralization. *Software and System Modeling*, 13(2):599–619, 2014.

[Fil12] H.-G. Fill. Using obfuscating transformations for supporting the sharing and analysis of conceptual models. In Susanne Robra-Bissantz and Dirk Mattfeld, editors, *Multikonferenz Wirtschaftsinformatik 2012 - Teilkonferenz Modellierung betrieblicher Informationssysteme*, Braunschweig, 2012. GITO Verlag.

[FJ12] Ales Frece and Matjaz B. Juric. Modeling functional requirements for configurable content- and context-aware dynamic service selection in business process models. *Journal of Visual Languages & Computing*, 23(4):223 – 247, 2012.

[FY10] Z. Fang and C. Yin. Bpm architecture design based on cloud computing. In *Intelligent Information Management, Vol. 2 No. 5*, pages 329–333, 2010.

[FYG09] Walid Fdhila, Ustun Yildiz, and Claude Godart. A flexible approach for automatic process decentralization using dependency tables. In *IEEE International Conference on Web Services, ICWS 2009, Los Angeles, CA, USA, 6-10 July 2009*, pages 847–855, 2009.

[Gam88] Diego Gambetta. *Trust: Making and Breaking Cooperative Relations*, volume 52. Blackwell, 1988.

[GANYG15] Elio Goettelmann, Amina Ahmed-Nacer, Samir Youcef, and Claude Godart. Paving the way towards semi-automatic design-time business process model obfuscation. In *Web Services (ICWS), 2015 IEEE International Conference on*, June 2015.

[Gar13] Gartner. Gartner says worldwide public cloud services market to total $131 billion. `http://www.gartner.com/newsroom/id/2352816`, 2013. [Online ; accessed 21-May-2013].

[GBO+07] Tyrone Grandison, Marcel Bilger, L. O'Connor, Marcel Graf, Morton Swimmer, Matthias Schunter, Andreas Wespi, and Nev Zunic. Elevating the discussion on security management: The data centric paradigm. In *Proceedings of BDIM 2007, 2nd IEEE/IFIP International Workshop on Business-Driven IT Management, May 21, 2007, Munich, Germany*, pages 84–93, 2007.

[GDG+14] Elio Goettelmann, Karim Dahman, Benjamin Gâteau, Eric Dubois, and Claude Godart. A security risk assessment model for business process deployment in the cloud. In *IEEE International Conference on Services Computing, SCC 2014, Anchorage, AK, USA, June 27 - July 2, 2014*, pages 307–314, 2014.

[GDGG14] Elio Goettelmann, Karim Dahman, Benjamin Gâteau, and Claude Godart. A formal broker framework for secure and cost-effective business process deployment on multiple clouds. In *Information Systems Engineering in Complex Environments - CAiSE Forum 2014, Thessaloniki, Greece, June 16-20, 2014, Selected Extended Papers*, pages 3–19, 2014.

[Gen09] Craig Gentry. Fully homomorphic encryption using ideal lattices. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31 - June 2, 2009*, pages 169–178, 2009.

[GFG13] Elio Goettelmann, Walid Fdhila, and Claude Godart. Partitioning and cloud deployment of composite web services under security constraints. In *2013 IEEE International Conference on Cloud Engineering, IC2E 2013, San Francisco, CA, USA, March 25-27, 2013*, pages 193–200, 2013.

[GL97] Fred Glover and Manuel Laguna. *Tabu Search*. Kluwer Academic Publishers, Norwell, MA, USA, 1997.

[GMG13] Elio Goettelmann, Nicolas Mayer, and Claude Godart. A general approach for a trusted deployment of a business process in clouds. In *Fifth International Conference on Management of Emergent Digital EcoSystems, MEDES '13, Luxembourg, Luxembourg, October 29-31, 2013*, pages 92–99, 2013.

[GMG14] Elio Goettelmann, Nicolas Mayer, and Claude Godart. Integrating security risk management into business process management for the cloud. In *IEEE 16th Conference on Business Informatics, CBI 2014, Geneva, Switzerland, July 14-17, 2014 - Volume 1*, pages 86–93, 2014.

[GMR+12] Nelson Gonzalez, Charles Miers, Fernando Redigolo, Marcos Simplicio, Tereza Carvalho, Mats Näslund, and Makan Pourzandi. A quantitative analysis of current security concerns and solutions for cloud computing. *Journal of Cloud Computing*, 2012.

[HLOS06] Shawn Hernan, Scott Lambert, Tomasz Ostwald, and Adam Shostack. Uncover security design flaws using the stride approach. `https://msdn.microsoft.com/en-us/magazine/cc163519.aspx`, 2006. [Online ; accessed 02-April-2015].

[HM79] Ching-Lai Hwang and Abu Syed Md. Masud. Methods for multiple objective decision making. In *Multiple Objective Decision Making — Methods and Applications*, volume 164 of *Lecture Notes in Economics and Mathematical Systems*, pages 21–283. Springer Berlin Heidelberg, 1979.

[Hul08] Richard Hull. Artifact-centric business process models: Brief survey of research results and challenges. In Robert Meersman and Zahir Tari, editors, *On the Move to Meaningful Internet Systems: OTM 2008*, volume 5332 of *Lecture Notes in Computer Science*, pages 1152–1163. Springer Berlin Heidelberg, 2008.

[ISO11] ISO/IEC 27005, Information tech., Security techniques, Information security risk management, 2011.

[ISO15] ISO/IEC 27017, Information tech., Security techniques, Code of practice for information security controls for cloud computing services based on ISO/IEC 27002, 2015. Status: under development.

[JLW+11] Jiulei Jiang, Jiajin Le, Yan Wang, Jie Sun, and Feng He. The bpm architecture based on cloud computing. In *Knowledge Acquisition and Modeling (KAM), 2011 Fourth International Symposium on*, pages 196–198, Oct 2011.

[JSB+11] M. Jensen, J. Schwenk, J. Bohli, N. Gruschka, and L.L. Iacono. Security prospects through cloud computing by adopting multiple clouds. In *CLOUD'11*, pages 565 –572, 2011.

[Jür02] Jan Jürjens. Umlsec: Extending uml for secure systems development. In Jean-Marc Jézéquel, Heinrich Hussmann, and Stephen Cook, editors, *UML 2002 – The Unified Modeling Language*, volume 2460 of *Lecture Notes in Computer Science*, pages 412–425. Springer Berlin Heidelberg, 2002.

[KH07] Pawan Khera and Bill Hefley. eSourcing capability model for client organizations (eSCM-CL) annotated bibliography. *SSRN Journal*, 2007.

[KLW08] Santhosh Kumaran, Rong Liu, and Frederick Wu. On the duality of information-centric and activity-centric models of business processes. In *Advanced Information Systems Engineering*, volume 5074 of *Lecture Notes in Computer Science*, pages 32–47. Springer Berlin Heidelberg, 2008.

[KP76] Brian W. Kernighan and P. L. Plauger. *Software Tools*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1976.

[KR01] Konstantin Knorr and Susanne Röhrig. Security requirements of e-business processes. In Beat Schmid, Katarina Stanoevska-Slabeva, and Volker Tschammer, editors, *Towards the E-Society*, volume 74 of *IFIP International Federation for Information Processing*, pages 72–86. Springer US, 2001.

[Lin09] David Linthicum. Soa cloud computing relationship leaves some folks in a fog. `http://gcn.com/Articles/2009/03/09/Guest-commentary-SOA-cloud.aspx`, 2009. [Online ; accessed 03-July-2015].

[LTM+11] Fang Liu, Jin Tong, Jian Mao, Rober Bohn, John Messina, Lee Badger, and Dawn Leaf. NIST cloud computing reference architecture. Technical report, National Institute of Standards and Technology (NIST), 2011.

[MAA11] Carlos Monsalve, Alain April, and Alain Abran. Requirements elicitation using bpm notations: Focusing on the strategic level representation. ACACOS'11, pages 235–241, 2011.

[May09] Nicolas Mayer. *Model-based Management of Information System Security Risk*. PhD thesis, University of Namur, Apr 2009.

[MF02] Peter Merz and Bernd Freisleben. Greedy and local search heuristics for unconstrained binary quadratic programming. *Journal of Heuristics*, 8(2):197–213, 2002.

[MF12] David Martinho and Diogo R. Ferreira. Securely storing and executing business processes in the cloud. In *Business Process Management Workshops - BPM 2012 International Workshops, Tallinn, Estonia, September 3, 2012. Revised Papers*, pages 707–712, 2012.

[MG11] Peter Mell and Timothy Grance. The NIST definition of cloud computing. Technical report, National Institute of Standards and Technology (NIST), 2011.

[MH06] Michael Zur Muehlen and Danny Ting-Yi Ho. Risk management in the bpm lifecycle. In Christoph J. Bussler and Armin Haller, editors, *Business Process Management Workshops*, volume 3812 of *Lecture Notes in Computer Science*, pages 454–466. Springer Berlin Heidelberg, 2006.

[MH12] Jean-Henry Morin and Anat Hovav. Strategic value and drivers behind organizational adoption of enterprise drm: The korean case. *Journal of Service Science Research*, 4(1):143–168, 2012.

[MJ10] Vinod Muthusamy and Hans-Arno Jacobsen. Bpm in cloud architectures: Business process management with slas and events. In Richard Hull, Jan Mendling, and Stefan Tai, editors, *Business Process Management*, volume 6336 of *Lecture Notes in Computer Science*, pages 5–10. Springer Berlin Heidelberg, 2010.

[MLB+11] Sean Marston, Zhi Li, Subhajyoti Bandyopadhyay, Juheng Zhang, and Anand Ghalsasi. Cloud computing - the business perspective. *Decision Support System*, Apr 2011.

[MMGG03] Haralambos Mouratidis, Gordon A. Manson, Abdullah Gani, and Paolo Giorgini. Analysing security requirements of information systems using tropos. Angers, France, 2003.

[MMM+08] Raimundas Matulevicius, Nicolas Mayer, Haralambos Mouratidis, Eric Dubois, Patrick Heymans, and Nicolas Genon. Adapting secure tropos for security risk management in the early phases of information systems development. In *Advanced Information Systems Engineering, 20th International Conference, CAISE 2008, Montpellier, France, June 16-20, 2008, Proceedings*, pages 541–555, 2008.

[MPR+09] Gerald Münzl, Bernhard Przywra, Martin Reti, Jörg Schäfer, Karin Sondermann, Mathias Weber, and Andreas Wilker. Cloud computing - evolution in der technik, revolution im business. Technical report, 2009.

[MR14] Juergen Mangler and Stefanie Rinderle-Ma. CPEE - cloud process execution engine. In *Proceedings of the BPM Demo Sessions 2014 Co-located with the 12th International Conference on Business Process Management (BPM 2014), Eindhoven, The Netherlands, September 10, 2014.*, page 51, 2014.

[MS95] Salvatore T. March and Gerald F. Smith. Design and natural science research on information technology. *Decis. Support Syst.*, 15(4):251–266, dec 1995.

[MSSN04] J. Mendling, M. Strembeck, G. Stermsek, and G. Neumann. An approach to extract rbac models from bpel4ws processes. In *Enabling Technologies: Infrastructure for Collaborative Enterprises, 2004. WET ICE 2004. 13th IEEE International Workshops on*, pages 81–86, June 2004.

[NIS02] Information security - guide for conducting risk assessments. Technical report, National Institute of Standards and Technology, 2002.

[OAS07] Web Services Business Process Execution Language (WSBPEL) 2.0. Technical report, Organization for the Advancement of Structured Information Standards (OASIS), 2007.

[OAS09] Web Services Reliable Messaging (WS-ReliableMessaging). Technical report, Organization for the Advancement of Structured Information Standards (OASIS), 2009.

[OBG13] Wendpanga Francis Ouedraogo, Frédérique Biennier, and Parisa Ghodous. Model driven security in a multi-cloud context. *IJEBM*, 11(3), 2013.

[OMG11] Business Process Model and Notation (BPMN) 2.0. Technical report, Object Management Group (OMG), 2011.

[OMG13] Unified Modelling Language (UML) 2.5. Technical report, Object Management Group (OMG), 2013.

[PGBD12] Artem Polyvyanyy, Luciano García-Bañuelos, and Marlon Dumas. Structuring acyclic process models. *Information Systems*, 37(6):518 – 538, 2012. BPM 2010.

[PGPM12] Elda Paja, Paolo Giorgini, Stéphane Paul, and PerHåkon Meland. Security requirements engineering for secure business processes. In Laila Niedrite, Renate Strazdina, and Benkt Wangler, editors, *Workshops on Business Informatics Research*, volume 106 of *Lecture Notes in Business Information Processing*, pages 77–89. Springer Berlin Heidelberg, 2012.

[PPKW11] M. Pathirage, S. Perera, I. Kumara, and S. Weerawarana. A multi-tenant architecture for business process executions. In *Web Services (ICWS), 2011 IEEE International Conference on*, pages 121–128, July 2011.

[PRZB12] Raluca A. Popa, Catherine M. S. Redfield, Nickolai Zeldovich, and Hari Balakrishnan. Cryptdb: processing queries on an encrypted database. *Commun. ACM*, 55(9):103–111, 2012.

[RFMP07] Alfonso Rodríguez, Eduardo Fernández-Medina, and Mario Piattini. A bpmn extension for the modeling of security requirements in business processes. *IEICE - Trans. Inf. Syst.*, E90-D(4):745–752, Mar 2007.

[Row14] Robert D. Rowley. Professional social networking. *Current Psychiatry Reports*, 16(12), 2014.

[Sch10] Lutz Schubert. The future of cloud computing - opportunities for european cloud computing beyond 2012. Technical report, European Commission - Information Society and Media, 2010.

[SLK09] Stefan Sackmann, Lutz Lowis, and Kai Kittel. A risk based approach for selecting services in business process execution. In *Wirtschaftsinformatik (1)*, pages 357–366, 2009.

[Sta08] Richard Stallman. Cloud computing is a trap, warns GNU founder Richard Stallman. `http://www.guardian.co.uk/technology/2008/sep/29/cloud.computing.richard.stallman`, 2008. [Online ; accessed 21-May-2013].

[TBCB12] Sameh Hbaieb Turki, Farah Bellaaj, Anis Charfi, and Rafik Bouaziz. Modeling security requirements in service based business processes. In *Enterprise, Business-Process and Information Systems Modeling - 13th International Conference, BPMDS 2012, 17th International Conference, EMMSAD 2012, and 5th EuroSymposium, held at CAiSE 2012, Gdańsk,Poland, June 25-26, 2012. Proceedings*, pages 76–90, 2012.

[TJG+11] S. Tjoa, S. Jakoubi, G. Goluch, G. Kitzler, S. Goluch, and G. Quirchmayr. A formal approach enabling risk-aware business process modeling and simulation. *Services Computing, IEEE Transactions on*, 4(2):153–166, Apr 2011.

[VRMCL08] Luis M. Vaquero, Luis Rodero-Merino, Juan Caceres, and Maik Lindner. A break in the clouds: Towards a cloud definition. *SIGCOMM Comput. Commun. Rev.*, 39(1):50–55, dec 2008.

[VVK08] Jussi Vanhatalo, Hagen Völzer, and Jana Koehler. The refined process structure tree. In Marlon Dumas, Manfred Reichert, and Ming-Chien Shan, editors, *Business Process Management*, volume 5240 of *Lecture Notes in Computer Science*, pages 100–115. Springer Berlin Heidelberg, 2008.

[Wat12] Paul Watson. A multi-level security model for partitioning workflows over federated clouds. *Journal of Cloud Computing*, 1(1), 2012.

[Wes12] Mathias Weske. *Business Process Management - Concepts, Languages, Architectures, 2nd Edition*. Springer, 2012.

[Wie09] Roel Wieringa. Design science as nested problem solving. In *Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology, DESRIST 2009, Philadelphia, Pennsylvania, USA, May 7-8, 2009*, 2009.

[Wik13a] Wikipedia. Cloud computing (en). `http://en.wikipedia.org/wiki/Cloud_computing`, 2013. [Online ; accessed 05-April-2013].

[Wik13b] Wikipedia. Cloud computing (fr). `http://fr.wikipedia.org/wiki/Cloud_computing`, 2013. [Online ; accessed 05-April-2013].

[WMM08] Christian Wolter, Michael Menzel, and Christoph Meinel. Modelling security goals in business processes. In *In Modellierung 2008, volume P-127 of LNI*, pages 201–216. Köllen, 2008.

[WMS+09] Christian Wolter, Michael Menzel, Andreas Schaad, Philip Miseldine, and Christoph Meinel. Model-driven business process security requirement specification. *Journal of Systems Architecture*, 55(4):211–223, 2009. Secure Service-Oriented Architectures (Special Issue on Secure SOA).

[Woz12] Steve Wozniak. Apple co-founder wozniak sees trouble in the cloud. http://www.google.com/hostednews/afp/article/ALeqM5h1p0LVc4iFZxbWlflFGgcHhbRNCQ, 2012. [Online ; accessed 21-May-2013].

[WWHJ12] Sven Wenzel, Christian Wessel, Thorsten Humberg, and Jan Jürjens. Securing processes for outsourcing into the cloud. In *CLOSER 2012 - Proceedings of the 2nd International Conference on Cloud Computing and Services Science, Porto, Portugal, 18 - 21 April, 2012*, pages 675–680, 2012.

[Zac99] Giorgos Zacharia. *Collaborative reputation mechanisms for online communities*. PhD thesis, Massachusetts Institute of Technology, 1999.

[ZSM+10] Wenchao Zhou, Micah Sherr, William R. Marczak, Zhuoyao Zhang, Tao Tao, Boon Thau Loo, and Insup Lee. Towards a data-centric view of cloud security. In *Proceedings of the Second International CIKM Workshop on Cloud Data Management, CloudDb 2010, Toronto, Ontario, Canada, October 30, 2010*, pages 25–32, 2010.

Appendix A

List of Cloud Security Alliance Controls

Drawn from the Consensus Assessments Initiative Questionnaire (CAIQ-2015) [58]

Application & Interface Security	
AIS-01 Application Security Applications and programming interfaces (APIs) shall be designed, developed, deployed and tested in accordance with leading industry standards (e.g., OWASP for web applications) and adhere to applicable legal, statutory, or regulatory compliance obligations.	
AIS-01.1	*Do you use industry standards (Build Security in Maturity Model [BSIMM] benchmarks, Open Group ACS Trusted Technology Provider Framework, NIST, etc.) to build in security for your Systems/Software Development Lifecycle (SDLC) ?*
AIS-01.2	*Do you use an automated source code analysis tool to detect security defects in code prior to production ?*
AIS-01.3	*Do you use manual source-code analysis to detect security defects in code prior to production ?*
AIS-01.4	*Do you verify that all of your software suppliers adhere to industry standards for Systems/Software Development Lifecycle (SDLC) security ?*
AIS-01.5	*(SaaS only) Do you review your applications for security vulnerabilities and address any issues prior to deployment to production ?*
AIS-02 Customer Access Requirements Prior to granting customers access to data, assets, and information systems, (removed all) identified security, contractual, and regulatory requirements for customer access shall be addressed.	
AIS-02.1	*Are all identified security, contractual and regulatory requirements for customer access contractually addressed and remediated prior to granting customers access to data, assets and information systems ?*
AIS-02.2	*Are all requirements and trust levels for customers' access defined and documented ?*
AIS-03 Data Integrity Data input and output integrity routines (i.e., reconciliation and edit checks) shall be implemented for application interfaces and databases to prevent manual or systematic processing errors, corruption of data, or misuse.	
AIS-03.1	*Are data input and output integrity routines (i.e., reconciliation and edit checks) implemented for application interfaces and databases to prevent manual or systematic processing errors or corruption of data ?*
AIS-04 Data Security / Integrity Policies and procedures shall be established and maintained in support of data security to include (confidentiality, integrity and availability) across multiple system interfaces, jurisdictions and business functions to prevent improper disclosure, alternation, or destruction.	
AIS-04.1	*Is your Data Security Architecture designed using an industry standard (e.g., CDSA, MULITSAFE, CSA Trusted Cloud Architectural Standard, FedRAMP, CAESARS) ?*
Audit Assurance & Compliance	
AAC-01 Audit Planning Audit plans shall be developed and maintained to address business process disruptions. Auditing plans shall focus on reviewing the effectiveness of the implementation of security operations. All audit activities must be agreed upon prior to executing any audits.	
AAC-01.1	*Do you produce audit assertions using a structured, industry accepted format (e.g., CloudAudit/A6 URI Ontology, CloudTrust, SCAP/CYBEX, GRC XML, ISACA's Cloud Computing Management Audit/Assurance Program, etc.) ?*

58. https://cloudsecurityalliance.org/download/consensus-assessments-initiative-questionnaire-v3-0-1/

AAC-02 Independent Audits

Independent reviews and assessments shall be performed at least annually to ensure that the organization addresses nonconformities of established policies, standards, procedures and compliance obligations.

AAC-02.1	*Do you allow tenants to view your SOC2/ISO 27001 or similar third-party audit or certification reports ?*
AAC-02.2	*Do you conduct network penetration tests of your cloud service infrastructure regularly as prescribed by industry best practices and guidance ?*
AAC-02.3	*Do you conduct application penetration tests of your cloud infrastructure regularly as prescribed by industry best practices and guidance ?*
AAC-02.4	*Do you conduct internal audits regularly as prescribed by industry best practices and guidance ?*
AAC-02.5	*Do you conduct external audits regularly as prescribed by industry best practices and guidance ?*
AAC-02.6	*Are the results of the penetration tests available to tenants at their request ?*
AAC-02.7	*Are the results of internal and external audits available to tenants at their request ?*
AAC-02.8	*Do you have an internal audit program that allows for cross-functional audit of assessments ?*

AAC-03 Information System Regulatory Mapping

Organizations shall create and maintain a control framework which captures standards, regulatory, legal, and statutory requirements relevant for their business needs. The control framework shall be reviewed at least annually to ensure changes that could affect the business processes are reflected.

AAC-03.1	*Do you have the ability to logically segment or encrypt customer data such that data may be produced for a single tenant only, without inadvertently accessing another tenant's data ?*
AAC-03.2	*Do you have capability to recover data for a specific customer in the case of a failure or data loss ?*
AAC-03.3	*Do you have the capability to restrict the storage of customer data to specific countries or geographic locations ?*
AAC-03.4	*Do you have a program in place that includes the ability to monitor changes to the regulatory requirements in relevant jurisdictions, adjust your security program for changes to legal requirements, and ensure compliance with relevant regulatory requirements ?*

Business Continuity Management & Operational Resilience

BCR-01 Business Continuity Planning

A consistent unified framework for business continuity planning and plan development shall be established, documented and adopted to ensure all business continuity plans are consistent in addressing priorities for testing, maintenance, and information security requirements. Requirements for business continuity plans include the following:

- Defined purpose and scope, aligned with relevant dependencies
- Accessible to and understood by those who will use them
- Owned by a named person(s) who is responsible for their review, update, and approval
- Defined lines of communication, roles, and responsibilities
- Detailed recovery procedures, manual work-around, and reference information
- Method for plan invocation

BCR-01.1	*Do you provide tenants with geographically resilient hosting options ?*
BCR-01.2	*Do you provide tenants with infrastructure service failover capability to other providers ?*

BCR-02 Business Continuity Testing

Business continuity and security incident response plans shall be subject to testing at planned intervals or upon significant organizational or environmental changes. Incident response plans shall involve impacted customers (tenant) and other business relationships that represent critical intra-supply chain business process dependencies.

BCR-02.1	*Are business continuity plans subject to test at planned intervals or upon significant organizational or environmental changes to ensure continuing effectiveness ?*

BCR-03 Power / Telecommunications

Datacenter utilities services and environmental conditions (e.g., water, power, temperature and humidity controls, telecommunications,and internet connectivity) shall be secured, monitored, maintained, and tested for continual effectiveness at planned intervals to ensure protection from unauthorized interception or damage, and designed with automated fail-over or other redundancies in the event of planned or unplanned disruptions.

BCR-03.1	*Do you provide tenants with documentation showing the transport route of their data between your systems ?*
BCR-03.2	*Can tenants define how their data is transported and through which legal jurisdictions ?*

BCR-04 Documentation

Information system documentation (e.g., administrator and user guides, and architecture diagrams) shall be made available to authorized personnel to ensure the following:

- Configuring, installing, and operating the information system
- Effectively using the system's security features

BCR-04.1 *Are information system documents (e.g., administrator and user guides, architecture diagrams, etc.) made available to authorized personnel to ensure configuration, installation and operation of the information system ?*

BCR-05 Environmental Risks

Physical protection against damage from natural causes and disasters, as well as deliberate attacks, including fire, flood, atmospheric electrical discharge, solar induced geomagnetic storm, wind, earthquake, tsunami, explosion, nuclear accident, volcanic activity, biological hazard, civil unrest, mudslide, tectonic activity, and other forms of natural or man-made disaster shall be anticipated, designed, and have countermeasures applied.

BCR-05.1 *Is physical protection against damage (e.g., natural causes, natural disasters, deliberate attacks) anticipated and designed with countermeasures applied ?*

BCR-06 Equipment Location

To reduce the risks from environmental threats, hazards, and opportunities for unauthorized access, equipment shall be kept away from locations subject to high probability environmental risks and supplemented by redundant equipment located at a reasonable distance.

BCR-06.1 *Are any of your data centers located in places that have a high probability /occurrence of high-impact environmental risks (floods, tornadoes, earthquakes, hurricanes, etc.) ?*

BCR-07 Equipment Maintenance

Policies and procedures shall be established, and supporting business processes and technical measures implemented, for equipment maintenance ensuring continuity and availability of operations and support personnel.

BCR-07.1 *If using virtual infrastructure, does your cloud solution include independent hardware restore and recovery capabilities ?*

BCR-07.2 *If using virtual infrastructure, do you provide tenants with a capability to restore a Virtual Machine to a previous state in time ?*

BCR-07.3 *If using virtual infrastructure, do you allow virtual machine images to be downloaded and ported to a new cloud provider ?*

BCR-07.4 *If using virtual infrastructure, are machine images made available to the customer in a way that would allow the customer to replicate those images in their own off-site storage location ?*

BCR-07.5 *Does your cloud solution include software/provider independent restore and recovery capabilities ?*

BCR-08 Equipment Power Failures

Protection measures shall be put into place to react to natural and man-made threats based upon a geographically-specific Business Impact Assessment

BCR-08.1 *Are security mechanisms and redundancies implemented to protect equipment from utility service outages (e.g., power failures, network disruptions, etc.) ?*

BCR-09 Impact Analysis

There shall be a defined and documented method for determining the impact of any disruption to the organization (cloud provider, cloud consumer) that must incorporate the following:

- Identify critical products and services
- Identify all dependencies, including processes, applications, business partners, and third party service providers
- Understand threats to critical products and services
- Determine impacts resulting from planned or unplanned disruptions and how these vary over time
- Establish the maximum tolerable period for disruption
- Establish priorities for recovery
- Establish recovery time objectives for resumption of critical products and services within their maximum tolerable period of disruption
- Estimate the resources required for resumption

BCR-09.1 *Do you provide tenants with ongoing visibility and reporting of your operational Service Level Agreement (SLA) performance ?*

BCR-09.2 *Do you make standards-based information security metrics (CSA, CAMM, etc.) available to your tenants ?*

BCR-09.3 *Do you provide customers with ongoing visibility and reporting of your SLA performance ?*

BCR-10 Policy
Policies and procedures shall be established, and supporting business processes and technical measures implemented, for appropriate IT governance and service management to ensure appropriate planning, delivery and support of the organization's IT capabilities supporting business functions, workforce, and/or customers based on industry acceptable standards (i.e., ITIL v4 and COBIT 5). Additionally, policies and procedures shall include defined roles and responsibilities supported by regular workforce training.

BCR-10.1	*Are policies and procedures established and made available for all personnel to adequately support services operations' roles ?*

BCR-11 Retention Policy
Policies and procedures shall be established, and supporting business processes and technical measures implemented, for defining and adhering to the retention period of any critical asset as per established policies and procedures, as well as applicable legal, statutory, or regulatory compliance obligations. Backup and recovery measures shall be incorporated as part of business continuity planning and tested accordingly for effectiveness.

BCR-11.1	*Do you have technical control capabilities to enforce tenant data retention policies ?*
BCR-11.2	*Do you have a documented procedure for responding to requests for tenant data from governments or third parties ?*
BCR-11.4	*Have you implemented backup or redundancy mechanisms to ensure compliance with regulatory, statutory, contractual or business requirements ?*
BCR-11.5	*Do you test your backup or redundancy mechanisms at least annually ?*

Change Control & Configuration Management

CCC-01 New Development / Acquisition
Policies and procedures shall be established, and supporting business processes and technical measures implemented, to ensure the development and/or acquisition of new data, physical or virtual applications, infrastructure network and systems components, or any corporate, operations and/or datacenter facilities have been pre-authorized by the organization's business leadership or other accountable business role or function.

CCC-01.1	*Are policies and procedures established for management authorization for development or acquisition of new applications, systems, databases, infrastructure, services, operations and facilities ?*
CCC-01.2	*Is documentation available that describes the installation, configuration and use of products/services/features ?*

CCC-02 Outsourced Development
External business partners shall adhere to the same policies and procedures for change management, release, and testing as internal developers within the organization (e.g. ITIL service management processes).

CCC-02.1	*Do you have controls in place to ensure that standards of quality are being met for all software development ?*
CCC-02.2	*Do you have controls in place to detect source code security defects for any outsourced software development activities ?*

CCC-03 Quality Testing
Organization shall follow a defined qualty change control and testing process (e.g. ITIL Service Management) with established baselines, testing and release standards which focus on system availability, confidentiality and integrity of systems and services

CCC-03.1	*Do you provide your tenants with documentation that describes your quality assurance process ?*
CCC-03.2	*Is documentation describing known issues with certain products/services available ?*
CCC-03.3	*Are there policies and procedures in place to triage and remedy reported bugs and security vulnerabilities for product and service offerings ?*
CCC-03.4	*Are mechanisms in place to ensure that all debugging and test code elements are removed from released software versions ?*

CCC-04 Unauthorized Software Installations
Policies and procedures shall be established, and supporting business processes and technical measures implemented, to restrict the installation of unauthorized software on organizationally-owned or managed user end-point devices (e.g., issued workstations, laptops, and mobile devices) and IT infrastructure network and systems components.

CCC-04.1	*Do you have controls in place to restrict and monitor the installation of unauthorized software onto your systems ?*

CCC-05 Production Changes
Policies and procedures shall be established for managing the risks associated with applying changes to business-critical or customer (tenant) impacting (physical and virtual) application and system-system interface (API) designs and configurations, as well as infrastructure network and systems components. Technical measures shall be implemented to provide assurance that, prior to deployment, all changes directly correspond to a registered change request, business-critical or customer (tenant) , and/or authorization by, the customer (tenant) as per agreement (SLA).

CCC-05.1	*Do you provide tenants with documentation that describes your production change management procedures and their roles/rights/responsibilities within it ?*

Data Security & Information Lifecycle Management	
DSI-01 Classification	
Data and objects containing data shall be assigned a classification by the data owner based on data type, value, sensitivity, and criticality to the organization.	
DSI-01.1	*Do you provide a capability to identify virtual machines via policy tags/metadata (e.g., tags can be used to limit guest operating systems from booting/instantiating/transporting data in the wrong country) ?*
DSI-01.2	*Do you provide a capability to identify hardware via policy tags/metadata/hardware tags (e.g., TXT/TPM, VN-Tag, etc.) ?*
DSI-01.3	*Do you have a capability to use system geographic location as an authentication factor ?*
DSI-01.4	*Can you provide the physical location/geography of storage of a tenant's data upon request ?*
DSI-01.5	*Can you provide the physical location/geography of storage of a tenant's data in advance ?*
DSI-01.6	*Do you follow a structured data-labeling standard (e.g., ISO 15489, Oasis XML Catalog Specification, CSA data type guidance) ?*
DSI-01.7	*Do you allow tenants to define acceptable geographical locations for data routing or resource instantiation ?*
DSI-02 Data Inventory / Flows	
Policies and procedures shall be established to inventory, document, and maintain data flows for data that is resident (permanently or temporarily) within the service's applications and infrastructure network and systems. In particular, providers shall ensure that data that is subject to geographic residency requirements not be migrated beyond its defined bounds.	
DSI-02.1	*Do you inventory, document, and maintain data flows for data that is resident (permanent or temporary) within the services' applications and infrastructure network and systems ?*
DSI-02.2	*Can you ensure that data does not migrate beyond a defined geographical residency ?*
DSI-03 eCommerce Transactions	
Data related to electronic commerce (e-commerce) that traverses public networks shall be appropriately classified and protected from fraudulent activity, unauthorized disclosure, or modification in such a manner to prevent contract dispute and compromise of data.	
DSI-03.1	*Do you provide open encryption methodologies (3.4ES, AES, etc.) to tenants in order for them to protect their data if it is required to move through public networks (e.g., the Internet) ?*
DSI-03.2	*Do you utilize open encryption methodologies any time your infrastructure components need to communicate with each other via public networks (e.g., Internet-based replication of data from one environment to another) ?*
DSI-04 Handling / Labeling / Security Policy	
Policies and procedures shall be established for labeling, handling, and the security of data and objects which contain data. Mechanisms for label inheritance shall be implemented for objects that act as aggregate containers for data.	
DSI-04.1	*Are policies and procedures established for labeling, handling and the security of data and objects that contain data ?*
DSI-04.2	*Are mechanisms for label inheritance implemented for objects that act as aggregate containers for data ?*
DSI-05 Nonproduction Data	
Production data shall not be replicated or used in non-production environments.	
DSI-05.1	*Do you have procedures in place to ensure production data shall not be replicated or used in non-production environments ?*
DSI-06 Ownership / Stewardship	
All data shall be designated with stewardship, with assigned responsibilities defined, documented, and communicated.	
DSI-06.1	*Are the responsibilities regarding data stewardship defined, assigned, documented and communicated ?*
DSI-07 Secure Disposal	
Any use of customer data in non-production environments requires explicit, documented approval from all customers whose data is affected, and must comply with all legal and regulatory requirements for scrubbing of sensitive data elements.	
DSI-07.1	*Do you support secure deletion (e.g., degaussing/cryptographic wiping) of archived and backed-up data as determined by the tenant ?*
DSI-07.2	*Can you provide a published procedure for exiting the service arrangement, including assurance to sanitize all computing resources of tenant data once a customer has exited your environment or has vacated a resource ?*
Datacenter Security	
DCS-01 Asset Management	
Assets must be classified in terms of business criticality, service-level expectations, and operational continuity requirements. A complete inventory of business-critical assets located at all sites and/or geographical locations and their usage over time shall be maintained and updated regularly, and assigned ownership y defined roles and responsibilities.	
DCS-01.1	*Do you maintain a complete inventory of all of your critical assets that includes ownership of the asset ?*
DCS-01.2	*Do you maintain a complete inventory of all of your critical supplier relationships ?*

DCS-02 Controlled Access Points Physical security perimeters (e.g., fences, walls, barriers, guards, gates, electronic surveillance, physical authentication mechanisms, reception desks, and security patrols) shall be implemented to safeguard sensitive data and information systems.	
DCS-02.1	*Are physical security perimeters (e.g., fences, walls, barriers, guards, gates, electronic surveillance, physical authentication mechanisms, reception desks and security patrols) implemented ?*
DCS-03 Equipment Identification Automated equipment identification shall be used as a method of connection authentication. Location-aware technologies may be used to validate connection authentication integrity based on known equipment location.	
DCS-03.1	*Is automated equipment identification used as a method to validate connection authentication integrity based on known equipment location ?*
DCS-04 Offsite Authorization Authorization must be obtained prior to relocation or transfer of hardware, software, or data to an offsite premises.	
DCS-04.1	*Do you provide tenants with documentation that describes scenarios in which data may be moved from one physical location to another ? (e.g., offsite backups, business continuity failovers, replication)*
DCS-05 Offsite equipment Policies and procedures shall be established for the secure disposal of equipment (by asset type) used outside the organization's premise. This shall include a wiping solution or destruction process that renders recovery of information impossible. The erasure shall consist of a full write of the drive to ensure that the erased drive is released to inventory for reuse and deployment or securely stored until it can be destroyed.	
DCS-05.1	*Can you provide tenants with evidence documenting your policies and procedures governing asset management and repurposing of equipment ?*
DCS-06 Policy Policies and procedures shall be established, and supporting business processes implemented, for maintaining a safe and secure working environment in offices, rooms, facilities, and secure areas.	
DCS-06.1	*Can you provide evidence that policies, standards and procedures have been established for maintaining a safe and secure working environment in offices, rooms, facilities and secure areas ?*
DCS-06.2	*Can you provide evidence that your personnel and involved third parties have been trained regarding your documented policies, standards and procedures ?*
DCS-07 Secure Area Authorization Ingress and egress to secure areas shall be constrained and monitored by physical access control mechanisms to ensure that only authorized personnel are allowed access.	
DCS-07.1	*Do you allow tenants to specify which of your geographic locations their data is allowed to move into/out of (to address legal jurisdictional considerations based on where data is stored vs. accessed) ?*
DCS-08 Unauthorized Persons Entry Ingress and egress points such as service areas and other points where unauthorized personnel may enter the premises shall be monitored, controlled and, if possible, isolated from data storage and processing facilities to prevent unauthorized data corruption, compromise, and loss.	
DCS-08.1	*Are ingress and egress points, such as service areas and other points where unauthorized personnel may enter the premises, monitored, controlled and isolated from data storage and process ?*
DCS-09 User Access Physical access to information assets and functions by users and support personnel shall be restricted.	
DCS-09.1	*Do you restrict physical access to information assets and functions by users and support personnel ?*
Encryption & Key Management	
EKM-01 Entitlement Keys must have identifiable owners (binding keys to identities) and there shall be key management policies.	
EKM-01.1	*Do you have key management policies binding keys to identifiable owners ?*

EKM-02 Key Generation

Policies and procedures shall be established for the management of cryptographic keys in the service's cryptosystem (e.g., lifecycle management from key generation to revocation and replacement, public key infrastructure, cryptographic protocol design and algorithms used, access controls in place for secure key generation, and exchange and storage including segregation of keys used for encrypted data or sessions). Upon request, provider shall inform the customer (tenant) of changes within the cryptosystem, especially if the customer (tenant) data is used as part of the service, and/or the customer (tenant) has some shared responsibility over implementation of the control.

EKM-02.1	*Do you have a capability to allow creation of unique encryption keys per tenant ?*
EKM-02.2	*Do you have a capability to manage encryption keys on behalf of tenants ?*
EKM-02.3	*Do you maintain key management procedures ?*
EKM-02.4	*Do you have documented ownership for each stage of the lifecycle of encryption keys ?*
EKM-02.5	*Do you utilize any third party/open source/proprietary frameworks to manage encryption keys ?*

EKM-03 Encryption

Policies and procedures shall be established, and supporting business processes and technical measures implemented, for the use of encryption protocols for protection of sensitive data in storage (e.g., file servers, databases, and end-user workstations) and data in transmission (e.g., system interfaces, over public networks, and electronic messaging) as per applicable legal, statutory, and regulatory compliance obligations.

EKM-03.1	*Do you encrypt tenant data at rest (on disk/storage) within your environment ?*
EKM-03.2	*Do you leverage encryption to protect data and virtual machine images during transport across and between networks and hypervisor instances ?*
EKM-03.3	*Do you support tenant-generated encryption keys or permit tenants to encrypt data to an identity without access to a public key certificate (e.g. identity-based encryption) ?*
EKM-03.4	*Do you have documentation establishing and defining your encryption management policies, procedures and guidelines ?*

EKM-04 Storage and Access

Platform and data appropriate encryption (e.g., AES-256) in open/validated formats and standard algorithms shall be required. Keys shall not be stored in the cloud (i.e. at the cloud provider in question), but maintained by the cloud consumer or trusted key management provider. Key management and key usage shall be separated duties.

EKM-04.1	*Do you have platform and data appropriate encryption that uses open/validated formats and standard algorithms ?*
EKM-04.2	*Are your encryption keys maintained by the cloud consumer or a trusted key management provider ?*
EKM-04.3	*Do you store encryption keys in the cloud ?*
EKM-04.4	*Do you have separate key management and key usage duties ?*

Governance and Risk Management

GRM-01 Baseline Requirements

Baseline security requirements shall be established for developed or acquired, organizationally-owned or managed, physical or virtual, applications and infrastructure system and network components that comply with applicable legal, statutory and regulatory compliance obligations. Deviations from standard baseline configurations must be authorized following change management policies and procedures prior to deployment, provisioning, or use. Compliance with security baseline requirements must be reassessed at least annually unless an alternate frequency has been established and established and authorized based on business need.

GRM-01.1	*Do you have documented information security baselines for every component of your infrastructure (e.g., hypervisors, operating systems, routers, DNS servers, etc.) ?*
GRM-01.2	*Do you have a capability to continuously monitor and report the compliance of your infrastructure against your information security baselines ?*
GRM-01.3	*Do you allow your clients to provide their own trusted virtual machine image to ensure conformance to their own internal standards ?*

GRM-02 Risk Assessments

Risk assessments associated with data governance requirements shall be conducted at planned intervals and shall consider the following:

- Awareness of where sensitive data is stored and transmitted across applications, databases, servers, and network infrastructure
- Compliance with defined retention periods and end-of-life disposal requirements
- Data classification and protection from unauthorized use, access, loss, destruction, and falsification

GRM-02.1	*Do you provide security control health data in order to allow tenants to implement industry standard Continuous Monitoring (which allows continual tenant validation of your physical and logical control status) ?*
GRM-02.2	*Do you conduct risk assessments associated with data governance requirements at least once a year ?*

GRM-03 Management Oversight

Managers are responsible for maintaining awareness of, and complying with, security policies, procedures and standards that are relevant to their area of responsibility.

GRM-03.1 *Are your technical, business, and executive managers responsible for maintaining awareness of and compliance with security policies, procedures, and standards for both themselves and their employees as they pertain to the manager and employees' area of responsibility ?*

GRM-04 Management Program

An Information Security Management Program (ISMP) shall be developed, documented, approved, and implemented that includes administrative, technical, and physical safeguards to protect assets and data from loss, misuse, unauthorized access, disclosure, alteration, and destruction. The security program shall include, but not be limited to, the following areas insofar as they relate to the characteristics of the business:

- Risk management
- Security policy
- Organization of information security
- Asset management
- Human resources security
- Physical and environmental security
- Communications and operations management
- Access control
- Information systems acquisition, development, and maintenance

GRM-04.1 *Do you provide tenants with documentation describing your Information Security Management Program (ISMP) ?*

GRM-04.2 *Do you review your Information Security Management Program (ISMP) least once a year ?*

GRM-05 Management Support / Involvement

Executive and line management shall take formal action to support information security through clearly-documented direction and commitment, and shall ensure the action has been assigned.

GRM-05.1 *Do you ensure your providers adhere to your information security and privacy policies ?*

GRM-06 Policy

Information security policies and procedures shall be established and made readily available for review by all impacted personnel and external business relationships. Information security policies must be authorized by the organization's business leadership (or other accountable business role or function) and supported by a strategic business plan and an information security management program inclusive of defined information security roles and responsibilities for business leadership.

GRM-06.1 *Do your information security and privacy policies align with industry standards (ISO-27001, ISO-22307, CoBIT, etc.) ?*

GRM-06.2 *Do you have agreements to ensure your providers adhere to your information security and privacy policies ?*

GRM-06.3 *Can you provide evidence of due diligence mapping of your controls, architecture and processes to regulations and/or standards ?*

GRM-06.4 *Do you disclose which controls, standards, certifications and/or regulations you comply with ?*

GRM-07 Policy Enforcement

A formal disciplinary or sanction policy shall be established for employees who have violated security policies and procedures. Employees shall be made aware of what action might be taken in the event of a violation, and disciplinary measures must be stated in the policies and procedures.

GRM-07.1 *Is a formal disciplinary or sanction policy established for employees who have violated security policies and procedures ?*

GRM-07.2 *Are employees made aware of what actions could be taken in the event of a violation via their policies and procedures ?*

GRM-08 Business / Policy Change Impacts

Risk assessment results shall include updates to security policies, procedures, standards, and controls to ensure that they remain relevant and effective.

GRM-08.1 *Do risk assessment results include updates to security policies, procedures, standards and controls to ensure they remain relevant and effective ?*

GRM-09 Policy Reviews

The organization's business leadership (or other accountable business role or function) shall review the information security policy at planned intervals or as a result of changes to the organization to ensure its continuing alignment with the security strategy, effectiveness, accuracy, relevance, and applicability to legal, statutory, or regulatory compliance obligations.

GRM-09.1 *Do you notify your tenants when you make material changes to your information security and/or privacy policies ?*

GRM-09.2 *Do you perform, at minimum, annual reviews to your privacy and security policies ?*

GRM-10 Assessments

Aligned with the enterprise-wide framework, formal risk assessments shall be performed at least annually or at planned intervals, (and in conjunction with any changes to information systems) to determine the likelihood and impact of all identified risks using qualitative and quantitative methods. The likelihood and impact associated with inherent and residual risk shall be determined independently, considering all risk categories (e.g., audit results, threat and vulnerability analysis, and regulatory compliance).

GRM-10.1	*Are formal risk assessments aligned with the enterprise-wide framework and performed at least annually, or at planned intervals, determining the likelihood and impact of all identified risks, using qualitative and quantitative methods ?*
GRM-10.2	*Is the likelihood and impact associated with inherent and residual risk determined independently, considering all risk categories (e.g., audit results, threat and vulnerability analysis, and regulatory compliance) ?*

GRM-11 Program

Organizations shall develop and maintain an enterprise risk management framework to mitigate risk to an acceptable level.

GRM-11.1	*Do you have a documented, organization-wide program in place to manage risk ?*
GRM-11.2	*Do you make available documentation of your organization-wide risk management program ?*

Human Resources

HRS-01 Asset Returns

Upon termination of workforce personnel and/or expiration of external business relationships, all organizationally-owned assets shall be returned within an established period.

HRS-01.1	*Are systems in place to monitor for privacy breaches and notify tenants expeditiously if a privacy event may have impacted their data ?*
HRS-01.2	*Is your Privacy Policy aligned with industry standards ?*

HRS-02 Background Screening

Pursuant to local laws, regulations, ethics, and contractual constraints, all employment candidates, contractors, and third parties shall be subject to background verification proportional to the data classification to be accessed, the business requirements, and acceptable risk.

HRS-02.1	*Pursuant to local laws, regulations, ethics and contractual constraints, are all employment candidates, contractors and involved third parties subject to background verification ?*

HRS-03 Employment Agreements

Employment agreements shall incorporate provisions and/or terms for adherence to established information governance and security policies and must be signed by newly hired or on-boarded workforce personnel (e.g., full or part-time employee or contingent staff) prior to granting workforce personnel user access to corporate facilities, resources, and assets.

HRS-03.1	*Do you specifically train your employees regarding their specific role and the information security controls they must fulfill ?*
HRS-03.2	*Do you document employee acknowledgment of training they have completed ?*
HRS-03.3	*Are all personnel required to sign NDA or Confidentiality Agreements as a condition of employment to protect customer/tenant information ?*
HRS-03.4	*Is successful and timed completion of the training program considered a prerequisite for acquiring and maintaining access to sensitive systems ?*
HRS-03.5	*Are personnel trained and provided with awareness programs at least once a year ?*

HRS-04 Employment Termination

Roles and responsibilities for performing employment termination or change in employment procedures shall be assigned, documented, and communicated.

HRS-04.1	*Are documented policies, procedures and guidelines in place to govern change in employment and/or termination ?*
HRS-04.2	*Do the above procedures and guidelines account for timely revocation of access and return of assets ?*

HRS-05 Portable / Mobile Devices

Policies and procedures shall be established, and supporting business processes and technical measures implemented, to manage business risks associated with permitting mobile device access to corporate resources and may require the implementation of higher assurance compensating controls and acceptable-use policies and procedures (e.g., mandated security training, stronger identity, entitlement and access controls, and device monitoring).

HRS-05.1	*Are policies and procedures established and measures implemented to strictly limit access to your sensitive data and tenant data from portable and mobile devices (e.g. laptops, cell phones and personal digital assistants (PDAs)), which are generally higher-risk than non-portable devices (e.g., desktop computers at the provider organization's facilities) ?*

HRS-06 Nondisclosure Agreements

Requirements for non-disclosure or confidentiality agreements reflecting the organization's needs for the protection of data and operational details shall be identified, documented, and reviewed at planned intervals.

HRS-06.1	*Are requirements for non-disclosure or confidentiality agreements reflecting the organization's needs for the protection of data and operational details identified, documented and reviewed at planned intervals ?*

HRS-07 Roles / Responsibilities

Roles and responsibilities of contractors, employees, and third-party users shall be documented as they relate to information assets and security.

HRS-07.1 *Do you provide tenants with a role definition document clarifying your administrative responsibilities versus those of the tenant ?*

HRS-08 Acceptable Use

Policies and procedures shall be established, and supporting business processes and technical measures implemented, for defining allowances and conditions for permitting usage of organizationally-owned or managed user end-point devices (e.g., issued workstations, laptops, and mobile devices) and IT infrastructure network and systems components. Additionally, defining allowances and conditions to permit usage of personal mobile devices and associated applications with access to corporate resources (i.e., BYOD) shall be considered and incorporated as appropriate.

HRS-08.1 *Do you provide documentation regarding how you may or access tenant data and metadata ?*

HRS-08.2 *Do you collect or create metadata about tenant data usage through inspection technologies (search engines, etc.) ?*

HRS-08.3 *Do you allow tenants to opt out of having their data/metadata accessed via inspection technologies ?*

HRS-09 Training / Awareness

A security awareness training program shall be established for all contractors, third-party users, and employees of the organization and mandated when appropriate. All individuals with access to organizational data shall receive appropriate awareness training and regular updates in organizational procedures, processes, and policies relating to their professional function relative to the organization.

HRS-09.1 *Do you provide a formal, role-based, security awareness training program for cloud-related access and data management issues (e.g., multi-tenancy, nationality, cloud delivery model segregation of duties implications and conflicts of interest) for all persons with access to tenant data ?*

HRS-09.2 *Are administrators and data stewards properly educated on their legal responsibilities with regard to security and data integrity ?*

HRS-10 User Responsibility

All personnel shall be made aware of their roles and responsibilities for:

- Maintaining awareness and compliance with established policies and procedures and applicable legal, statutory, or regulatory compliance obligations.
- Maintaining a safe and secure working environment

HRS-10.1 *Are users made aware of their responsibilities for maintaining awareness and compliance with published security policies, procedures, standards and applicable regulatory requirements ?*

HRS-10.2 *Are users made aware of their responsibilities for maintaining a safe and secure working environment ?*

HRS-10.3 *Are users made aware of their responsibilities for leaving unattended equipment in a secure manner ?*

HRS-11 Workspace

Policies and procedures shall be established to require that unattended workspaces do not have openly visible (e.g., on a desktop) sensitive documents and user computing sessions had been disabled after an established period of inactivity.

HRS-11.1 *Do your data management policies and procedures address tenant and service level conflicts of interests ?*

HRS-11.2 *Do your data management policies and procedures include a tamper audit or software integrity function for unauthorized access to tenant data ?*

HRS-11.3 *Does the virtual machine management infrastructure include a tamper audit or software integrity function to detect changes to the build/configuration of the virtual machine ?*

Identity & Access Management

IAM-01 Audit Tools Access

Access to, and use of, audit tools that interact with the organization's information systems shall be appropriately segmented and restricted to prevent compromise and misuse of log data.

IAM-01.1 *Do you restrict, log and monitor access to your information security management systems ? (E.g., hypervisors, firewalls, vulnerability scanners, network sniffers, APIs, etc.)*

IAM-01.2 *Do you monitor and log privileged access (administrator level) to information security management systems ?*

IAM-02 User Access Policy

User access policies and procedures shall be established, and supporting business processes and technical measures implemented, for ensuring appropriate identity, entitlement, and access management for all internal corporate and customer (tenant) users with access to data and organizationally-owned or managed (physical and virtual) application interfaces and infrastructure network and systems components. These policies, procedures, processes, and measures must incorporate the following:

- Procedures and supporting roles and responsibilities for provisioning and de-provisioning user account entitlements following the rule of least privilege based on job function (e.g., internal employee and contingent staff personnel changes, customer-controlled access, suppliers' business relationships, or other third-party business relationships)
- Business case considerations for higher levels of assurance and multi-factor authentication secrets (e.g., management interfaces, key generation, remote access, segregation of duties, emergency access, large-scale provisioning or geographically-distributed deployments, and personnel redundancy for critical systems)
- Access segmentation to sessions and data in multi-tenant architectures by any third party (e.g., provider and/or other customer (tenant))
- Identity trust verification and service-to-service application (API) and information processing interoperability (e.g., SSO and federation)
- Account credential lifecycle management from instantiation through revocation
- Account credential and/or identity store minimization or re-use when feasible
- Authentication, authorization, and accounting (AAA) rules for access to data and sessions (e.g., encryption and strong/multi-factor, expireable, non-shared authentication secrets)
- Permissions and supporting capabilities for customer (tenant) controls over authentication, authorization, and accounting (AAA) rules for access to data and sessions

Adherence to applicable legal, statutory, or regulatory compliance requirements

IAM-02.1 *Do you have controls in place ensuring timely removal of systems access that is no longer required for business purposes ?*

IAM-02.2 *Do you provide metrics to track the speed with which you are able to remove systems access that is no longer required for business purposes ?*

IAM-03 Diagnostic / Configuration Ports Access

User access to diagnostic and configuration ports shall be restricted to authorized individuals and applications.

IAM-03.1 *Do you use dedicated secure networks to provide management access to your cloud service infrastructure ?*

IAM-04 Policies and Procedures

Policies and procedures shall be established to store and manage identity information about every person who accesses IT infrastructure and to determine their level of access. Policies shall also be developed to control access to network resources based on user identity.

IAM-04.1 *Do you manage and store the identity of all personnel who have access to the IT infrastructure, including their level of access ?*

IAM-04.2 *Do you manage and store the user identity of all personnel who have network access, including their level of access ?*

IAM-05 Segregation of Duties

User access policies and procedures shall be established, and supporting business processes and technical measures implemented, for restricting user access as per defined segregation of duties to address business risks associated with a user-role conflict of interest.

IAM-05.1 *Do you provide tenants with documentation on how you maintain segregation of duties within your cloud service offering ?*

IAM-06 Source Code Access Restriction

Access to the organization's own developed applications, program, or object source code, or any other form of intellectual property (IP), and use of proprietary software shall be appropriately restricted following the rule of least privilege based on job function as per established user access policies and procedures.

IAM-06.1 *Are controls in place to prevent unauthorized access to your application, program or object source code, and assure it is restricted to authorized personnel only ?*

IAM-06.2 *Are controls in place to prevent unauthorized access to tenant application, program or object source code, and assure it is restricted to authorized personnel only ?*

IAM-07 Third Party Access

The identification, assessment, and prioritization of risks posed by business processes requiring third-party access to the organization's information systems and data shall be followed by coordinated application of resources to minimize, monitor, and measure likelihood and impact of unauthorized or inappropriate access. Compensating controls derived from the risk analysis shall be implemented prior to provisioning access.

IAM-07.1	*Do you provide multi-failure disaster recovery capability ?*
IAM-07.2	*Do you monitor service continuity with upstream providers in the event of provider failure ?*
IAM-07.3	*Do you have more than one provider for each service you depend on ?*
IAM-07.4	*Do you provide access to operational redundancy and continuity summaries, including the services you depend on ?*
IAM-07.5	*Do you provide the tenant the ability to declare a disaster ?*
IAM-07.6	*Do you provided a tenant-triggered failover option ?*
IAM-07.7	*Do you share your business continuity and redundancy plans with your tenants ?*

IAM-08 User Access Restriction / Authorization

Policies and procedures are established for permissible storage and access of identities used for authentication to ensure identities are only accessible based on rules of least privilege and replication limitation only to users explicitly defined as business necessary.

IAM-08.1	*Do you document how you grant and approve access to tenant data ?*
IAM-08.2	*Do you have a method of aligning provider and tenant data classification methodologies for access control purposes ?*

IAM-09 User Access Authorization

Provisioning user access (e.g., employees, contractors, customers (tenants), business partners and/or supplier relationships) to data and organizationally-owned or managed (physical and virtual) applications, infrastructure systems, and network components shall be authorized by the organization's management prior to access being granted and appropriately restricted as per established policies and procedures. Upon request, provider shall inform customer (tenant) of this user access, especially if customer (tenant) data is used as part of the service and/or customer (tenant) has some shared responsibility over implementation of control.

IAM-09.1	*Does your management provision the authorization and restrictions for user access (e.g. employees, contractors, customers (tenants), business partners and/or suppliers) prior to their access to data and any owned or managed (physical and virtual) applications, infrastructure systems and network components ?*
IAM-09.2	*Do your provide upon request user access (e.g. employees, contractors, customers (tenants), business partners and/or suppliers) to data and any owned or managed (physical and virtual) applications, infrastructure systems and network components ?*

IAM-10 User Access Reviews

User access shall be authorized and revalidated for entitlement appropriateness, at planned intervals, by the organization's business leadership or other accountable business role or function supported by evidence to demonstrate the organization is adhering to the rule of least privilege based on job function. For identified access violations, remediation must follow established user access policies and procedures.

IAM-10.1	*Do you require at least annual certification of entitlements for all system users and administrators (exclusive of users maintained by your tenants) ?*
IAM-10.2	*If users are found to have inappropriate entitlements, are all remediation and certification actions recorded ?*
IAM-10.3	*Will you share user entitlement remediation and certification reports with your tenants, if inappropriate access may have been allowed to tenant data ?*

IAM-11 User Access Revocation

Timely de-provisioning (revocation or modification) of user access to data and organizationally-owned or managed (physical and virtual) applications, infrastructure systems, and network components, shall be implemented as per established policies and procedures and based on user's change in status (e.g., termination of employment or other business relationship, job change or transfer). Upon request, provider shall inform customer (tenant) of these changes, especially if customer (tenant) data is used as part the service and/or customer (tenant) has some shared responsibility over implementation of control.

IAM-11.1	*Is timely deprovisioning, revocation or modification of user access to the organizations systems, information assets and data implemented upon any change in status of employees, contractors, customers, business partners or involved third parties ?*
IAM-11.2	*Is any change in user access status intended to include termination of employment, contract or agreement, change of employment or transfer within the organization ?*

IAM-12 User ID Credentials

Internal corporate or customer (tenant) user account credentials shall be restricted as per the following, ensuring appropriate identity, entitlement, and access management and in accordance with established policies and procedures:

- Identity trust verification and service-to-service application (API) and information processing interoperability (e.g., SSO and Federation)
- Account credential lifecycle management from instantiation through revocation
- Account credential and/or identity store minimization or re-use when feasible
- Adherence to industry acceptable and/or regulatory compliant authentication, authorization, and accounting (AAA) rules (e.g., strong/multi-factor, expireable, non-shared authentication secrets)

IAM-12.1	*Do you support use of, or integration with, existing customer-based Single Sign On (SSO) solutions to your service ?*
IAM-12.2	*Do you use open standards to delegate authentication capabilities to your tenants ?*
IAM-12.3	*Do you support identity federation standards (SAML, SPML, WS-Federation, etc.) as a means of authenticating/authorizing users ?*
IAM-12.4	*Do you have a Policy Enforcement Point capability (e.g., XACML) to enforce regional legal and policy constraints on user access ?*
IAM-12.5	*Do you have an identity management system (enabling classification of data for a tenant) in place to enable both role-based and context-based entitlement to data ?*
IAM-12.6	*Do you provide tenants with strong (multifactor) authentication options (digital certs, tokens, biometrics, etc.) for user access ?*
IAM-12.7	*Do you allow tenants to use third-party identity assurance services ?*
IAM-12.8	*Do you support password (minimum length, age, history, complexity) and account lockout (lockout threshold, lockout duration) policy enforcement ?*
IAM-12.9	*Do you allow tenants/customers to define password and account lockout policies for their accounts ?*
IAM-12.10	*Do you support the ability to force password changes upon first logon ?*
IAM-12.11	*Do you have mechanisms in place for unlocking accounts that have been locked out (e.g., self-service via email, defined challenge questions, manual unlock) ?*

IAM-13 Utility Programs Access

Utility programs capable of potentially overriding system, object, network, virtual machine, and application controls shall be restricted.

IAM-13.1	*Are utilities that can significantly manage virtualized partitions (e.g., shutdown, clone, etc.) appropriately restricted and monitored ?*
IAM-13.2	*Do you have a capability to detect attacks that target the virtual infrastructure directly (e.g., shimming, Blue Pill, Hyper jumping, etc.) ?*
IAM-13.3	*Are attacks that target the virtual infrastructure prevented with technical controls ?*

Infrastructure & Virtualization Security

IVS-01 Audit Logging / Intrusion Detection

Higher levels of assurance are required for protection, retention, and lifecyle management of audit logs, adhering to applicable legal, statutory or regulatory compliance obligations and providing unique user access accountability to detect potentially suspicious network behaviors and/or file integrity anomalies, and to support forensic investigative capabilities in the event of a security breach.

IVS-01.1	*Are file integrity (host) and network intrusion detection (IDS) tools implemented to help facilitate timely detection, investigation by root cause analysis and response to incidents ?*
IVS-01.2	*Is physical and logical user access to audit logs restricted to authorized personnel ?*
IVS-01.3	*Can you provide evidence that due diligence mapping of regulations and standards to your controls/architecture/processes has been done ?*
IVS-01.4	*Are audit logs centrally stored and retained ?*
IVS-01.5	*Are audit logs reviewed on a regular basis for security events (e.g., with automated tools) ?*

IVS-02 Change Detection

The provider shall ensure the integrity of all virtual machine images at all times. Any changes made to virtual machine images must be logged and an alert raised regardless of their running state (e.g. dormant, off, or running). The results of a change or move of an image and the subsequent validation of the image's integrity must be immediately available to customers through electronic methods (e.g. portals or alerts).

IVS-02.1	*Do you log and alert any changes made to virtual machine images regardless of their running state (e.g. dormant, off or running) ?*
IVS-02.2	*Are changes made to virtual machines, or moving of an image and subsequent validation of the image's integrity, made immediately available to customers through electronic methods (e.g. portals or alerts) ?*

IVS-03 Clock Synchronization

A reliable and mutually agreed upon external time source shall be used to synchronize the system clocks of all relevant information processing systems to facilitate tracing and reconstitution of activity timelines.

IVS-03.1	*Do you use a synchronized time-service protocol (e.g., NTP) to ensure all systems have a common time reference ?*

IVS-04 Capacity / Resource Planning

The availability, quality, and adequate capacity and resources shall be planned, prepared, and measured to deliver the required system performance in accordance with legal, statutory, and regulatory compliance obligations. Projections of future capacity requirements shall be made to mitigate the risk of system overload.

IVS-04.1	*Do you provide documentation regarding what levels of system (network, storage, memory, I/O, etc.) oversubscription you maintain and under what circumstances/scenarios ?*
IVS-04.2	*Do you restrict use of the memory oversubscription capabilities present in the hypervisor ?*
IVS-04.3	*Do your system capacity requirements take into account current, projected and anticipated capacity needs for all systems used to provide services to the tenants ?*
IVS-04.4	*Is system performance monitored and tuned in order to continuously meet regulatory, contractual and business requirements for all the systems used to provide services to the tenants ?*

IVS-05 Management - Vulnerability Management

Implementers shall ensure that the security vulnerability assessment tools or services accommodate the virtualization technologies used (e.g. virtualization aware).

IVS-05.1	*Do security vulnerability assessment tools or services accommodate the virtualization technologies being used (e.g. virtualization aware) ?*

IVS-06 Network Security

Network environments and virtual instances shall be designed and configured to restrict and monitor traffic between trusted and untrusted connections, these configurations shall be reviewed at least annually, and supported by a documented justification for use for all allowed services, protocols, and ports, and compensating controls.

IVS-06.1	*For your IaaS offering, do you provide customers with guidance on how to create a layered security architecture equivalence using your virtualized solution ?*
IVS-06.2	*Do you regularly update network architecture diagrams that include data flows between security domains/zones ?*
IVS-06.3	*Do you regularly review for appropriateness the allowed access/connectivity (e.g., firewall rules) between security domains/zones within the network ?*
IVS-06.4	*Are all firewall access control lists documented with business justification ?*

IVS-07 OS Hardening and Base Conrols

Each operating system shall be hardened to provide only necessary ports, protocols, and services to meet business needs and have in place supporting technical controls such as: antivirus, file integrity monitoring, and logging as part of their baseline operating build standard or template.

IVS-07.1	*Are operating systems hardened to provide only the necessary ports, protocols and services to meet business needs using technical controls (i.e antivirus, file integrity monitoring and logging) as part of their baseline build standard or template ?*

IVS-08 Production / Nonproduction Environments

Production and non-production environments shall be separated to prevent unauthorized access or changes to information assets. Separation of the environments may include: stateful inspection firewalls, domain/realm authentication sources, and clear segregation of duties for personnel accessing these environments as part of their job duties.

IVS-08.1	*For your SaaS or PaaS offering, do you provide tenants with separate environments for production and test processes ?*
IVS-08.2	*For your IaaS offering, do you provide tenants with guidance on how to create suitable production and test environments ?*
IVS-08.3	*Do you logically and physically segregate production and non-production environments ?*

IVS-09 Segmentation

Multi-tenant organizationally-owned or managed (physical and virtual) applications, and infrastructure system and network components, shall be designed, developed, deployed and configured such that provider and customer (tenant) user access is appropriately segmented from other tenant users, based on the following considerations:

- Established policies and procedures
- Isolation of business critical assets and/or sensitive user data and sessions that mandate stronger internal controls and high levels of assurance
- Compliance with legal, statutory and regulatory compliance obligations

IVS-09.1	*Are system and network environments protected by a firewall or virtual firewall to ensure business and customer security requirements ?*
IVS-09.2	*Are system and network environments protected by a firewall or virtual firewall to ensure compliance with legislative, regulatory and contractual requirements ?*
IVS-09.3	*Are system and network environments protected by a firewall or virtual firewall to ensure separation of production and non-production environments ?*
IVS-09.4	*Are system and network environments protected by a firewall or virtual firewall to ensure protection and isolation of sensitive data ?*

IVS-10 VM Security - vMotion Data Protection

Secured and encrypted communication channels shall be used when migrating physical servers, applications, or data to virtualized servers and, where possible, shall use a network segregated from production-level networks for such migrations.

IVS-10.1	*Are secured and encrypted communication channels used when migrating physical servers, applications or data to virtual servers ?*
IVS-10.2	*Do you use a network segregated from production-level networks when migrating physical servers, applications or data to virtual servers ?*

IVS-11 VMM Security - Hypervisor Hardening

Access to all hypervisor management functions or administrative consoles for systems hosting virtualized systems shall be restricted to personnel based upon the principle of least privilege and supported through technical controls (e.g., two-factor authentication, audit trails, IP address filtering, firewalls, and TLS encapsulated communications to the administrative consoles).

IVS-11.1	*Do you restrict personnel access to all hypervisor management functions or administrative consoles for systems hosting virtualized systems based on the principle of least privilege and supported through technical controls (e.g. two-factor authentication, audit trails, IP address filtering, firewalls and TLS-encapsulated communications to the administrative consoles) ?*

IVS-12 Wireless Security

Policies and procedures shall be established, and supporting business processes and technical measures implemented, to protect wireless network environments, including the following:

- Perimeter firewalls implemented and configured to restrict unauthorized traffic
- Security settings enabled with strong encryption for authentication and transmission, replacing vendor default settings (e.g., encryption keys, passwords, and SNMP community strings)
- User access to wireless network devices restricted to authorized personnel
- The capability to detect the presence of unauthorized (rogue) wireless network devices for a timely disconnect from the network

IVS-12.1	*Are policies and procedures established and mechanisms configured and implemented to protect the wireless network environment perimeter and to restrict unauthorized wireless traffic ?*
IVS-12.2	*Are policies and procedures established and mechanisms implemented to ensure wireless security settings are enabled with strong encryption for authentication and transmission, replacing vendor default settings ? (e.g., encryption keys, passwords, SNMP community strings)*
IVS-12.3	*Are policies and procedures established and mechanisms implemented to protect wireless network environments and detect the presence of unauthorized (rogue) network devices for a timely disconnect from the network ?*

IVS-13 Network Architecture

Network architecture diagrams shall clearly identify high-risk environments and data flows that may have legal compliance impacts. Technical measures shall be implemented and shall apply defense-in-depth techniques (e.g., deep packet analysis, traffic throttling, and black-holing) for detection and timely response to network-based attacks associated with anomalous ingress or egress traffic patterns (e.g., MAC spoofing and ARP poisoning attacks) and/or distributed denial-of-service (DDoS) attacks.

IVS-13.1	*Do your network architecture diagrams clearly identify high-risk environments and data flows that may have legal compliance impacts ?*
IVS-13.2	*Do you implement technical measures and apply defense-in-depth techniques (e.g., deep packet analysis, traffic throttling and black-holing) for detection and timely response to network-based attacks associated with anomalous ingress or egress traffic patterns (e.g., MAC spoofing and ARP poisoning attacks) and/or distributed denial-of-service (DDoS) attacks ?*

Interoperability & Portability

IPY-01 APIs

The provider shall use open and published APIs to ensure support for interoperability between components and to facilitate migrating applications.

IPY-01	*Do you publish a list of all APIs available in the service and indicate which are standard and which are customized ?*

IPY-02 Data Request

All structured and unstructured data shall be available to the customer and provided to them upon request in an industry-standard format (e.g., .doc, .xls, .pdf, logs, and flat files)

IPY-02	*Is unstructured customer data available on request in an industry-standard format (e.g., .doc, .xls, or .pdf) ?*

IPY-03 Policy & Legal

Policies, procedures, and mutually-agreed upon provisions and/or terms shall be established to satisfy customer (tenant) requirements for service-to-service application (API) and information processing interoperability, and portability for application development and information exchange, usage and integrity persistence.

IPY-03.1	*Do you provide policies and procedures (i.e. service level agreements) governing the use of APIs for interoperability between your service and third-party applications ?*
IPY-03.2	*Do you provide policies and procedures (i.e. service level agreements) governing the migration of application data to and from your service ?*

IPY-04 Standardized Network Protocols

The provider shall use secure (e.g., non-clear text and authenticated) standardized network protocols for the import and export of data and to manage the service, and shall make available a document to consumers (tenants) detailing the relevant interoperability and portability standards that are involved.

IPY-04.1	*Can data import, data export and service management be conducted over secure (e.g., non-clear text and authenticated), industry accepted standardized network protocols ?*
IPY-04.2	*Do you provide consumers (tenants) with documentation detailing the relevant interoperability and portability network protocol standards that are involved ?*

IPY-05 Virtualization
The provider shall use an industry-recognized virtualization platform and standard virtualization formats (e.g., OVF) to help ensure interoperability, and shall have documented custom changes made to any hypervisor in use, and all solution-specific virtualization hooks, available for customer review.

IPY-05.1	*Do you use an industry-recognized virtualization platform and standard virtualization formats (e,g., OVF) to help ensure interoperability ?*
IPY-05.2	*Do you have documented custom changes made to any hypervisor in use, and all solution-specific virtualization hooks available for customer review ?*

Mobile Security

MOS-01 Anti-Malware
Anti-malware awareness training, specific to mobile devices, shall be included in the provider's information security awareness training.

MOS-01	*Do you provide anti-malware training specific to mobile devices as part of your information security awareness training ?*

MOS-02 Application Stores
A documented list of approved application stores has been communicated as acceptable for mobile devices accessing or storing provider managed data.

MOS-02	*Do you document and make available lists of approved application stores for mobile devices accessing or storing company data and/or company systems ?*

MOS-03 Approved Applications
The company shall have a documented policy prohibiting the installation of non-approved applications or approved applications not obtained through a pre-identified application store.

MOS-03	*Do you have a policy enforcement capability (e.g., XACML) to ensure that only approved applications and those from approved application stores be loaded onto a mobile device ?*

MOS-04 Approved Software for BYOD
The BYOD policy and supporting awareness training clearly states the approved applications, application stores, and application extensions and plugins that may be used for BYOD usage.

MOS-04	*Does your BYOD policy and training clearly state which applications and applications stores are approved for use on BYOD devices ?*

MOS-05 Awareness and Training
The provider shall have a documented mobile device policy that includes a documented definition for mobile devices and the acceptable usage and requirements for all mobile devices. The provider shall post and communicate the policy and requirements through the company's security awareness and training program.

MOS-05	*Do you have a documented mobile device policy in your employee training that clearly defines mobile devices and the accepted usage and requirements for mobile devices ?*

MOS-06 Cloud Based Services
All cloud-based services used by the company's mobile devices or BYOD shall be pre-approved for usage and the storage of company business data.

MOS-06	*Do you have a documented list of pre-approved cloud based services that are allowed to be used for use and storage of company business data via a mobile device ?*

MOS-07 Compatibility
The company shall have a documented application validation process to test for mobile device, operating system, and application compatibility issues.

MOS-07	*Do you have a documented application validation process for testing device, operating system and application compatibility issues ?*

MOS-08 Device Eligibility
The BYOD policy shall define the device and eligibility requirements to allow for BYOD usage.

MOS-08	*Do you have a BYOD policy that defines the device(s) and eligibility requirements allowed for BYOD usage ?*

MOS-09 Device Inventory
An inventory of all mobile devices used to store and access company data shall be kept and maintained. All changes to the status of these devices, (i.e., operating system and patch levels, lost or decommissioned status, and to whom the device is assigned or approved for usage (BYOD), will be included for each device in the inventory.

MOS-09	*Do you maintain an inventory of all mobile devices storing and accessing company data which includes device status (os system and patch levels, lost or decommissioned, device assignee) ?*

MOS-10 Device Management
A centralized, mobile device management solution shall be deployed to all mobile devices permitted to store, transmit, or process customer data.

MOS-10	*Do you have a centralized mobile device management solution deployed to all mobile devices that are permitted to store, transmit, or process company data ?*

MOS-11 Encryption

The mobile device policy shall require the use of encryption either for the entire device or for data identified as sensitive on all mobile devices and shall be enforced through technology controls.

MOS-11 *Does your mobile device policy require the use of encryption for either the entire device or for data identified as sensitive enforceable through technology controls for all mobile devices ?*

MOS-12 Jailbreaking and Rooting

The mobile device policy shall prohibit the circumvention of built-in security controls on mobile devices (e.g. jailbreaking or rooting) and isenforced through detective and preventative controls on the device or through a centralized device management system (e.g. mobile device management).

MOS-12.1 *Does your mobile device policy prohibit the circumvention of built-in security controls on mobile devices (e.g., jailbreaking or rooting) ?*

MOS-12.2 *Do you have detective and preventative controls on the device or via a centralized device management system which prohibit the circumvention of built-in security controls ?*

MOS-13 Legal

The BYOD policy includes clarifying language for the expectation of privacy, requirements for litigation, e-discovery, and legal holds. The BYOD policy shall clearly state the expectations over the loss of non-company data in the case a wipe of the device is required.

MOS-13.1 *Does your BYOD policy clearly define the expectation of privacy, requirements for litigation, e-discovery and legal holds ?*

MOS-13.2 *Do you have detective and preventative controls on the device or via a centralized device management system which prohibit the circumvention of built-in security controls ?*

MOS-14 Lockout Screen

BYOD and/or company owned devices are configured to require an automatic lockout screen, and the requirement shall be enforced through technical controls.

MOS-14 *Do you require and enforce via technical controls an automatic lockout screen for BYOD and company owned devices ?*

MOS-15 Operating Systems

Changes to mobile device operating systems, patch levels, and/or applications shall be managed through the company's change management processes.

MOS-15 *Do you manage all changes to mobile device operating systems, patch levels and applications via your company's change management processes ?*

MOS-16 Passwords

Password policies, applicable to mobile devices, shall be documented and enforced through technical controls on all company devices or devices approved for BYOD usage, and shall prohibit the changing of password/PIN lengths and authentication requirements.

MOS-16.1 *Do you have password policies for enterprise issued mobile devices and/or BYOD mobile devices ?*

MOS-16.2 *Are your password policies enforced through technical controls (i.e. MDM) ?*

MOS-16.3 *Do your password policies prohibit the changing of authentication requirements (i.e. password/PIN length) via a mobile device ?*

MOS-17 Policy

The mobile device policy shall require the BYOD user to perform backups of data, prohibit the usage of unapproved application stores, and require the use of anti-malware software (where supported).

MOS-17.1 *Do you have a policy that requires BYOD users to perform backups of specified corporate data ?*

MOS-17.2 *Do you have a policy that requires BYOD users to prohibit the usage of unapproved application stores ?*

MOS-17.3 *Do you have a policy that requires BYOD users to use anti-malware software (where supported) ?*

MOS-18 Remote Wipe

All mobile devices permitted for use through the company BYOD program or a company-assigned mobile device shall allow for remote wipe by the company's corporate IT or shall have all company-provided data wiped by the company's corporate IT.

MOS-18.1 *Does your IT provide remote wipe or corporate data wipe for all company-accepted BYOD devices ?*

MOS-18.2 *Does your IT provide remote wipe or corporate data wipe for all company-assigned mobile devices ?*

MOS-19 Security Patches

Mobile devices connecting to corporate networks or storing and accessing company information shall allow for remote software version/patch validation. All mobile devices shall have the latest available security-related patches installed upon general release by the device manufacturer or carrier and authorized IT personnel shall be able to perform these updates remotely.

MOS-19.1 *Do your mobile devices have the latest available security-related patches installed upon general release by the device manufacturer or carrier ?*

MOS-19.2 *Do your mobile devices allow for remote validation to download the latest security patches by company IT personnel ?*

MOS-20 Users

The BYOD policy shall clarify the systems and servers allowed for use or access on a BYOD-enabled device.

MOS-20.1 *Does your BYOD policy clarify the systems and servers allowed for use or access on the BYOD-enabled device ?*

MOS-20.2 *Does your BYOD policy specify the user roles that are allowed access via a BYOD-enabled device ?*

Security Incident Management, E-Discovery & Cloud Forensics

SEF-01 Contact / Authority Maintenance

Points of contact for applicable regulation authorities, national and local law enforcement, and other legal jurisdictional authorities shall be maintained and regularly updated (e.g., change in impacted-scope and/or a change in any compliance obligation) to ensure direct compliance liaisons have been established and to be prepared for a forensic investigation requiring rapid engagement with law enforcement.

SEF-01.1 *Do you maintain liaisons and points of contact with local authorities in accordance with contracts and appropriate regulations ?*

SEF-02 Incident Management

Policies and procedures shall be established, and supporting business processes and technical measures implemented, to triage security-related events and ensure timely and thorough incident management, as per established IT service management policies and procedures.

SEF-02.1 *Do you have a documented security incident response plan ?*

SEF-02.2 *Do you integrate customized tenant requirements into your security incident response plans ?*

SEF-02.3 *Do you publish a roles and responsibilities document specifying what you vs. your tenants are responsible for during security incidents ?*

SEF-02.4 *Have you tested your security incident response plans in the last year ?*

SEF-03 Incident Reporting

Workforce personnel and external business relationships shall be informed of their responsibility and, if required, shall consent and/or contractually agree to report all information security events in a timely manner. Information security events shall be reported through predefined communications channels in a timely manner adhering to applicable legal, statutory, or regulatory compliance obligations.

SEF-03.1 *Does your security information and event management (SIEM) system merge data sources (app logs, firewall logs, IDS logs, physical access logs, etc.) for granular analysis and alerting ?*

SEF-03.2 *Does your logging and monitoring framework allow isolation of an incident to specific tenants ?*

SEF-04 Incident Response Legal Preparation

Proper forensic procedures, including chain of custody, are required for the presentation of evidence to support potential legal action subject to the relevant jurisdiction after an information security incident. Upon notification, customers and/or other external business partners impacted by a security breach shall be given the opportunity to participate as is legally permissible in the forensic investigation.

SEF-04.1 *Does your incident response plan comply with industry standards for legally admissible chain-of-custody management processes and controls ?*

SEF-04.2 *Does your incident response capability include the use of legally admissible forensic data collection and analysis techniques ?*

SEF-04.3 *Are you capable of supporting litigation holds (freeze of data from a specific point in time) for a specific tenant without freezing other tenant data ?*

SEF-04.4 *Do you enforce and attest to tenant data separation when producing data in response to legal subpoenas ?*

SEF-05 Incident Response Metrics

Mechanisms shall be put in place to monitor and quantify the types, volumes, and costs of information security incidents.

SEF-05.1 *Do you monitor and quantify the types, volumes and impacts on all information security incidents ?*

SEF-05.2 *Will you share statistical information for security incident data with your tenants upon request ?*

Supply Chain Management, Transparency and Accountability

STA-01 Data Quality and Integrity

Providers shall inspect, account for, and work with their cloud supply-chain partners to correct data quality errors and associated risks. Providers shall design and implement controls to mitigate and contain data security risks through proper separation of duties, role-based access, and least-privilege access for all personnel within their supply chain.

STA-01.1 *Do you inspect and account for data quality errors and associated risks, and work with your cloud supply-chain partners to correct them ?*

STA-01.2 *Do you design and implement controls to mitigate and contain data security risks through proper separation of duties, role-based access, and least-privileged access for all personnel within your supply chain ?*

STA-02 Incident Reporting

The provider shall make security incident information available to all affected customers and providers periodically through electronic methods (e.g. portals).

STA-02.1 *Do you make security incident information available to all affected customers and providers periodically through electronic methods (e.g. portals) ?*

STA-03 Network / Infrastructure Services

Business-critical or customer (tenant) impacting (physical and virtual) application and system-system interface (API) designs and configurations, and infrastructure network and systems components, shall be designed, developed, and deployed in accordance with mutually agreed-upon service and capacity-level expectations, as well as IT governance and service management policies and procedures.

STA-03.1 *Do you collect capacity and use data for all relevant components of your cloud service offering ?*

STA-03.2 *Do you provide tenants with capacity planning and use reports ?*

STA-04 Provider Internal Assessments

The provider shall perform annual internal assessments of conformance and effectiveness of its policies, procedures, and supporting measures and metrics.

STA-04.1 *Do you perform annual internal assessments of conformance and effectiveness of your policies, procedures, and supporting measures and metrics ?*

STA-05 Third Party Agreements

Supply chain agreements (e.g., SLAs) between providers and customers (tenants) shall incorporate at least the following mutually-agreed upon provisions and/or terms:

- Scope of business relationship and services offered (e.g., customer (tenant) data acquisition, exchange and usage, feature sets and functionality, personnel and infrastructure network and systems components for service delivery and support, roles and responsibilities of provider and customer (tenant) and any subcontracted or outsourced business relationships, physical geographical location of hosted services, and any known regulatory compliance considerations)
- Information security requirements, provider and customer (tenant) primary points of contact for the duration of the business relationship, and references to detailed supporting and relevant business processes and technical measures implemented to enable effectively governance, risk management, assurance and legal, statutory and regulatory compliance obligations by all impacted business relationships
- Notification and/or pre-authorization of any changes controlled by the provider with customer (tenant) impacts
- Timely notification of a security incident (or confirmed breach) to all customers (tenants) and other business relationships impacted (i.e., up- and down-stream impacted supply chain)
- Assessment and independent verification of compliance with agreement provisions and/or terms (e.g., industry-acceptable certification, attestation audit report, or equivalent forms of assurance) without posing an unacceptable business risk of exposure to the organization being assessed
- Expiration of the business relationship and treatment of customer (tenant) data impacted
- Customer (tenant) service-to-service application (API) and data interoperability and portability requirements for application development and information exchange, usage, and integrity persistence

STA-05.1 *Do you select and monitor outsourced providers in compliance with laws in the country where the data is processed, stored and transmitted ?*

STA-05.2 *Do you select and monitor outsourced providers in compliance with laws in the country where the data originates ?*

STA-05.3 *Does legal counsel review all third-party agreements ?*

STA-05.4 *Do third-party agreements include provision for the security and protection of information and assets ?*

STA-05.5 *Do you provide the client with a list and copies of all subprocessing agreements and keep this updated ?*

STA-06 Supply Chain Governance Reviews

Providers shall review the risk management and governance processes of their partners so that practices are consistent and aligned to account for risks inherited from other members of that partner's cloud supply chain.

STA-06.1 *Do you review the risk management and governanced processes of partners to account for risks inherited from other members of that partner's supply chain ?*

STA-07 Supply Chain Metrics

Policies and procedures shall be implemented to ensure the consistent review of service agreements (e.g., SLAs) between providers and customers (tenants) across the relevant supply chain (upstream/downstream). Reviews shall performed at least annually and identity non-conformance to established agreements. The reviews should result in actions to address service-level conflicts or inconsistencies resulting from disparate supplier relationships.

STA-07.1	*Are policies and procedures established, and supporting business processes and technical measures implemented, for maintaining complete, accurate and relevant agreements (e.g., SLAs) between providers and customers (tenants) ?*
STA-07.2	*Do you have the ability to measure and address non-conformance of provisions and/or terms across the entire supply chain (upstream/downstream) ?*
STA-07.3	*Can you manage service-level conflicts or inconsistencies resulting from disparate supplier relationships ?*
STA-07.4	*Do you review all agreements, policies and processes at least annually ?*

STA-08 Third Party Assessment

Providers shall assure reasonable information security across their information supply chain by performing an annual review. The review shall include all partners/third party providers upon which their information supply chain depends on.

STA-08.1	*Do you assure reasonable information security across your information supply chain by performing an annual review ?*
STA-08.2	*Does your annual review include all partners/third-party providers upon which your information supply chain depends ?*

STA-09 Third Party Audits

Third-party service providers shall demonstrate compliance with information security and confidentiality, access control, service definitions, and delivery level agreements included in third-party contracts. Third-party reports, records, and services shall undergo audit and review at least annually to govern and maintain compliance with the service delivery agreements.

STA-09.1	*Do you permit tenants to perform independent vulnerability assessments ?*
STA-09.2	*Do you have external third party services conduct vulnerability scans and periodic penetration tests on your applications and networks ?*

Threat and Vulnerbility Management

TVM-01 Antivirus / Malicious Software

Policies and procedures shall be established, and supporting business processes and technical measures implemented, to prevent the execution of malware on organizationally-owned or managed user end-point devices (i.e., issued workstations, laptops, and mobile devices) and IT infrastructure network and systems components.

TVM-01.1	*Do you have anti-malware programs that support or connect to your cloud service offerings installed on all of your systems ?*
TVM-01.2	*Do you ensure that security threat detection systems using signatures, lists or behavioral patterns are updated across all infrastructure components within industry accepted time frames ?*

TVM-02 Vulnerability / Patch Management

Policies and procedures shall be established, and supporting processes and technical measures implemented, for timely detection of vulnerabilities within organizationally-owned or managed applications, infrastructure network and system components (e.g. network vulnerability assessment, penetration testing) to ensure the efficiency of implemented security controls. A risk-based model for prioritizing remediation of identified vulnerabilities shall be used. Changes shall be managed through a change management process for all vendor-supplied patches, configuration changes, or changes to the organization's internally developed software. Upon request, the provider informs customer (tenant) of policies and procedures and identified weaknesses especially if customer (tenant) data is used as part the service and/or customer (tenant) has some shared responsibility over implementation of control.

TVM-02.1	*Do you conduct network-layer vulnerability scans regularly as prescribed by industry best practices ?*
TVM-02.2	*Do you conduct application-layer vulnerability scans regularly as prescribed by industry best practices ?*
TVM-02.3	*Do you conduct local operating system-layer vulnerability scans regularly as prescribed by industry best practices ?*
TVM-02.4	*Will you make the results of vulnerability scans available to tenants at their request ?*
TVM-02.5	*Do you have a capability to rapidly patch vulnerabilities across all of your computing devices, applications and systems ?*
TVM-02.6	*Will you provide your risk-based systems patching time frames to your tenants upon request ?*

TVM-03 Mobile Code

Policies and procedures shall be established, and supporting business processes and technical measures implemented, to prevent the execution of unauthorized mobile code, defined as software transferred between systems over a trusted or untrusted network and executed on a local system without explicit installation or execution by the recipient, on organizationally-owned or managed user end-point devices (e.g., issued workstations, laptops, and mobile devices) and IT infrastructure network and systems components.

TVM-03.1	*Is mobile code authorized before its installation and use, and the code configuration checked, to ensure that the authorized mobile code operates according to a clearly defined security policy ?*
TVM-03.2	*Is all unauthorized mobile code prevented from executing ?*

Appendix B

Multi-criteria optimization results

Overall thirty-two different runs were performed: four types of weightings settings, four types of threshold settings and two types of distributions (uniform and normal).

The graphs on the left always represent the uniform distribution, the graphs on the right the normal distribution. The two graphs at the top always show the *overall calculation time* (*calcTime* in milliseconds, left y-axis) for generating the full set of solutions and the *number of selected solutions* (*numSols*, right y-axis). The two graphs at the bottom show always the *exclusion ratio* (the percentage of excluded solutions from the initial set, *excRatio*, left y-axis) and the *average calculation time per solution* (the total calculation time brought to the number of solutions found, *avgTime* in milliseconds, right y-axis).

The different settings where defined as follows:

P1 The cost is the most important value, but is not sufficient to determine if a configuration is better or not. The cost and one other criterion (no matter which one) must be better in order to have a weighting-domination.

P2 The cost is still the most important value, and it still has to be combined with another criterion to create a weighting-domination. Except the complexity criterion, which even combined with the cost, cannot create this domination.

P3 The cost is the only criterion that creates the weighting-domination. All other criteria are disregarded and cannot create such a domination.

P4 All criteria are equally taken into account. Thus, to create a weighting-domination, there must be at least three criteria (no matter which ones) that are better. *E.g.*, costs and risks are not sufficient for generating such a domination.

T1 The different thresholds are globally "normal". Which means that we consider them as neither high nor low.

T2 Globally, the thresholds are lower than for T1. Thus it becomes a bit easier for a configuration to contradict the first domination criterion (the weighting-domination)

T3 In opposition to T2, here the thresholds are higher than for T1. Thus, it becomes more difficult for a configuration to contradict the first domination criterion

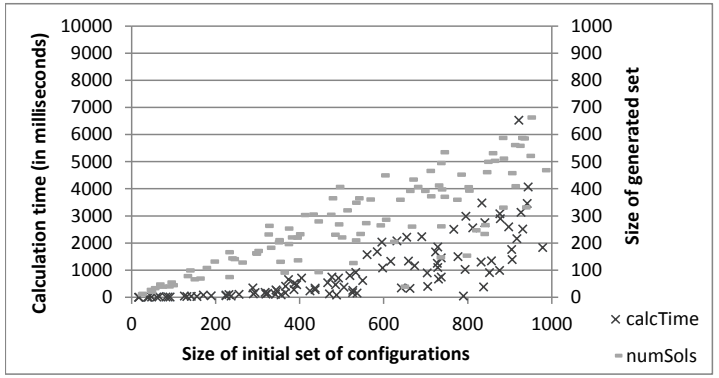
T4 Here we wanted to see the influence of disparate thresholds. They are globally lower, except for the QoS which is high.

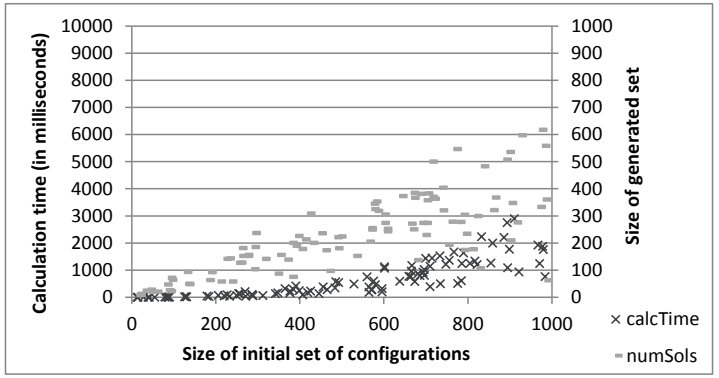
T1-P1	Cost	Risk	QoS	Complexity
Weighting	0.40	0.20	0.25	0.15
Threshold	100	1	2	0.2

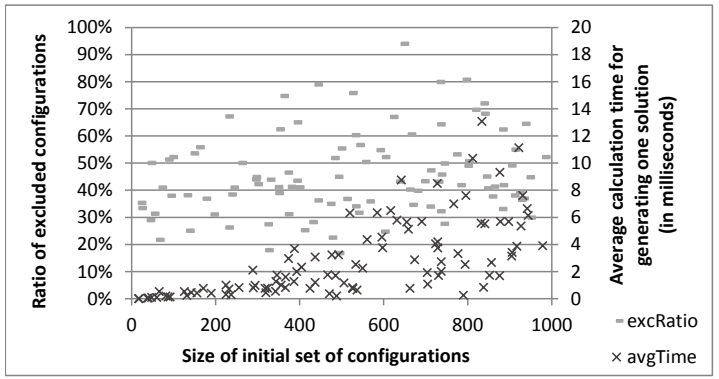
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	42.25%	14.86%	91.30%	0.157	0.025	UNIFORM
Time	2.426	0.00	11.34	2.408	5.796	

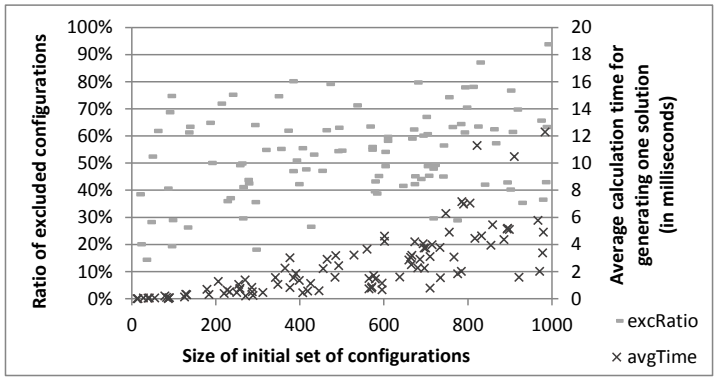
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	51.54%	18.92%	98.11%	0.145	0.021	NORMAL
Time	1.957	0.00	8.84	2.025	4.100	

T1-P2	Cost	Risk	QoS	Complexity
Weighting	0.40	0.20	0.30	0.10
Threshold	100	1	2	0.2









	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	45.36%	16.80%	93.93%	0.150	0.022	UNIFORM
Time	2.856	0.00	13.06	2.815	7.923	

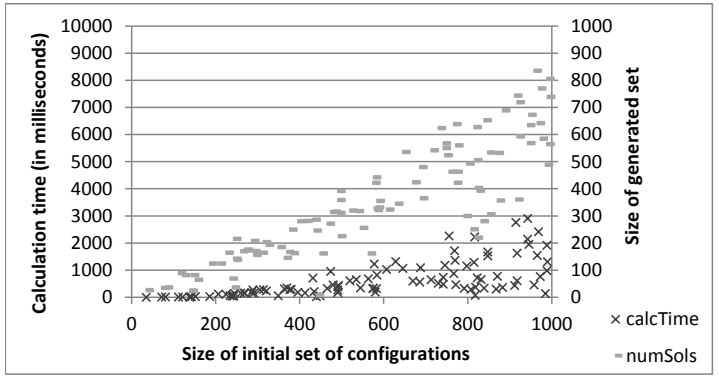
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	52.90%	14.29%	93.70%	0.161	0.026	NORMAL
Time	2.428	0.00	12.29	2.436	5.936	

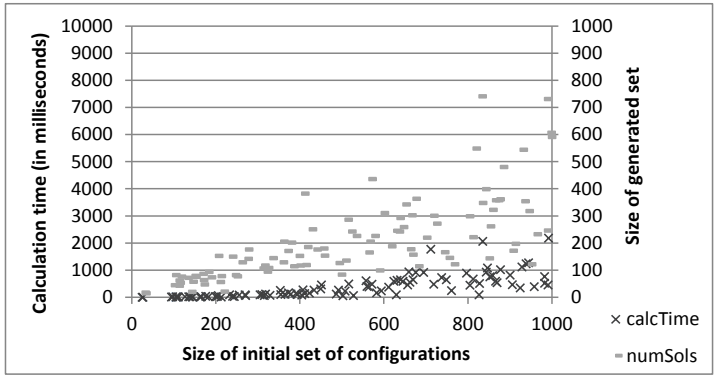
T1-P3	Cost	Risk	QoS	Complexity
Weighting	0.85	0.05	0.05	0.05
Threshold	100	1	2	0.2

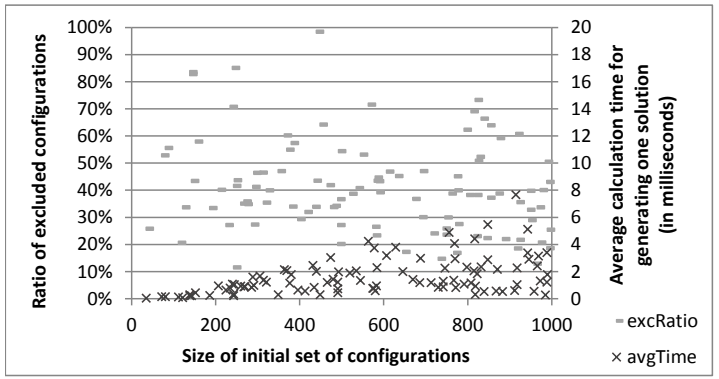
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	40.97%	4.76%	75.78%	0.142	0.020	UNIFORM
Time	2.544	0.00	18.29	2.840	8.068	

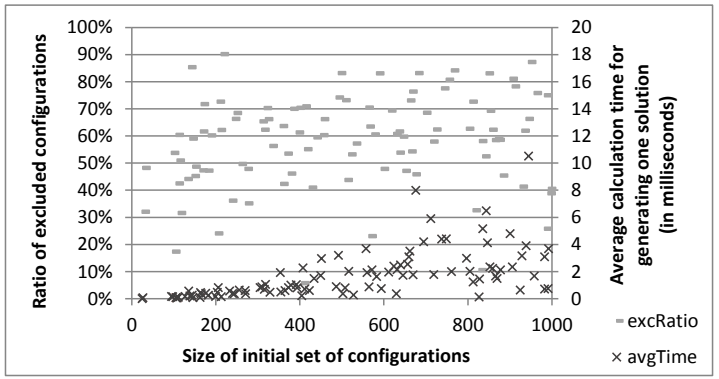
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	52.96%	14.58%	96.63%	0.160	0.026	NORMAL
Time	2.053	0.03	8.85	1.865	3.479	

T1-P4	Cost	Risk	QoS	Complexity
Weighting	0.25	0.25	0.25	0.25
Threshold	100	1	2	0.2









	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	40.85%	11.52%	98.41%	0.170	0.029	UNIFORM
Time	1.602	0.04	7.66	1.364	1.860	

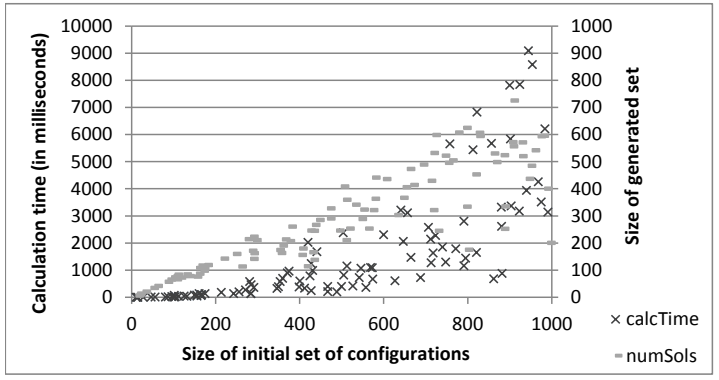
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	57.88%	5.68%	90.14%	0.170	0.029	NORMAL
Time	1.677	0.00	10.50	1.807	3.266	

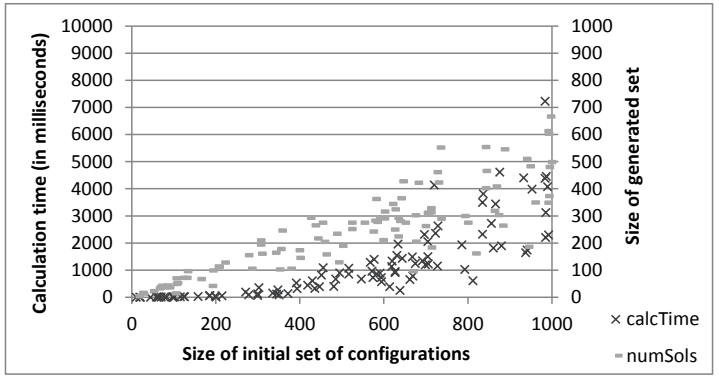
T2-P1	Cost	Risk	QoS	Complexity
Weighting	0.40	0.20	0.25	0.15
Threshold	100	1	1	0.1

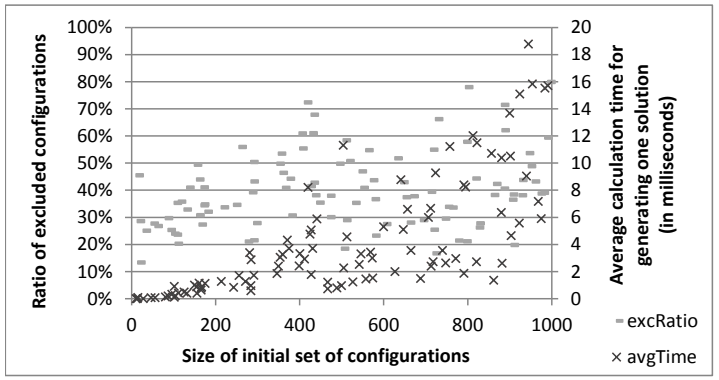
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	42.35%	10.65%	95.56%	0.149	0.022	UNIFORM
Time	4.429	0.10	19.40	4.732	22.393	

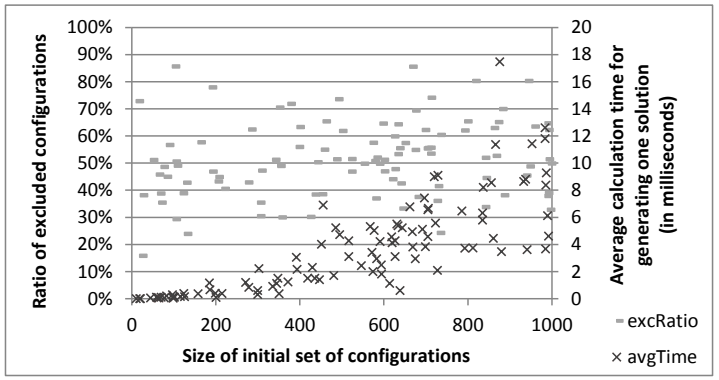
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	50.68%	15.00%	81.72%	0.134	0.018	NORMAL
Time	3.887	0.00	15.48	3.528	12.449	

T2-P2	Cost	Risk	QoS	Complexity
Weighting	0.40	0.20	0.30	0.10
Threshold	100	1	1	0.1









	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	39.73%	13.33%	79.82%	0.139	0.019	UNIFORM
Time	4.129	0.00	18.76	4.314	18.615	

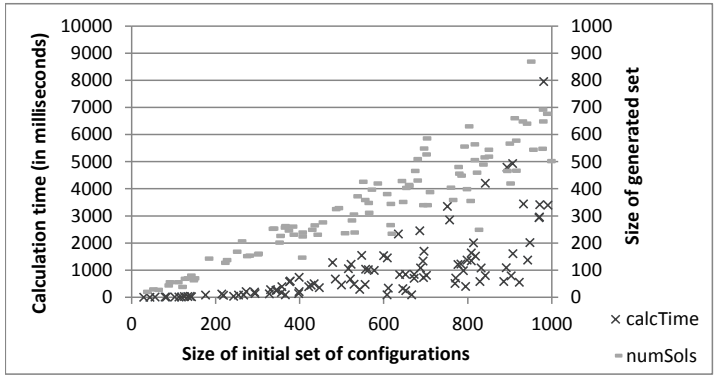
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	51.32%	15.79%	85.57%	0.139	0.019	NORMAL
Time	3.668	0.00	17.47	3.435	11.800	

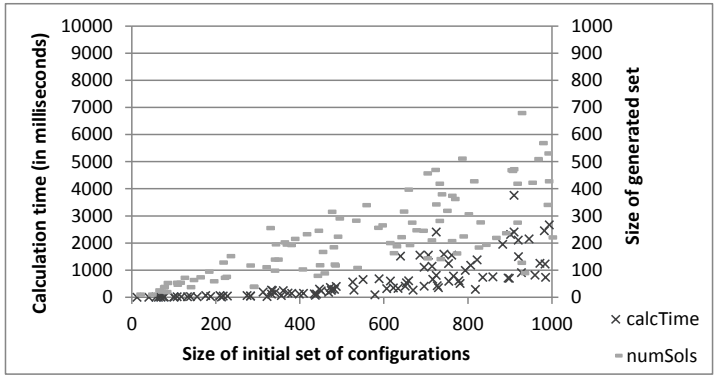
T2-P3	Cost	Risk	QoS	Complexity
Weighting	0.85	0.05	0.05	0.05
Threshold	100	1	1	0.1

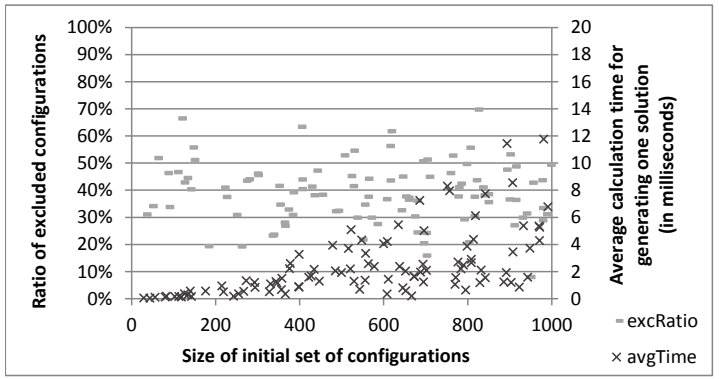
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	38.92%	9.09%	74.17%	0.134	0.018	UNIFORM
Time	5.479	0.00	22.89	4.898	23.986	

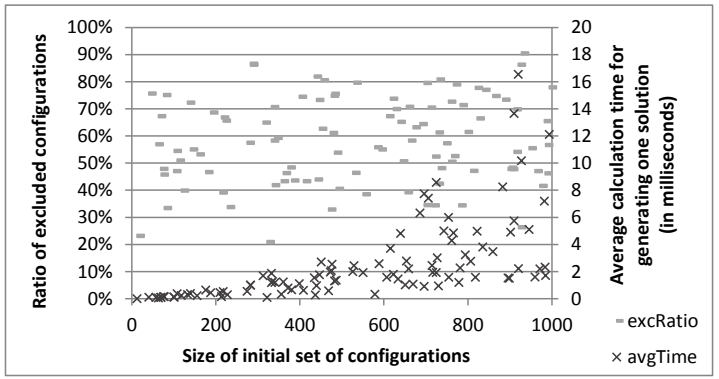
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	49.59%	14.29%	87.87%	0.158	0.025	NORMAL
Time	3.386	0.00	21.39	3.261	10.637	

T2-P4	Cost	Risk	QoS	Complexity
Weighting	0.25	0.25	0.25	0.25
Threshold	100	1	1	0.1





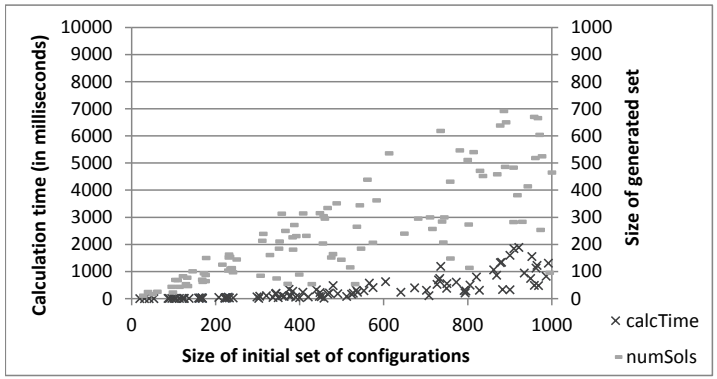


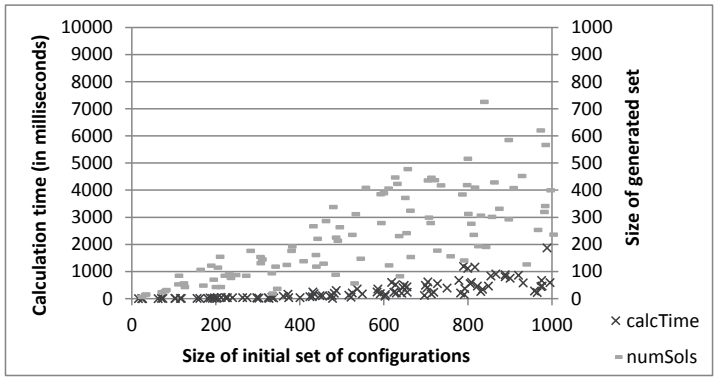


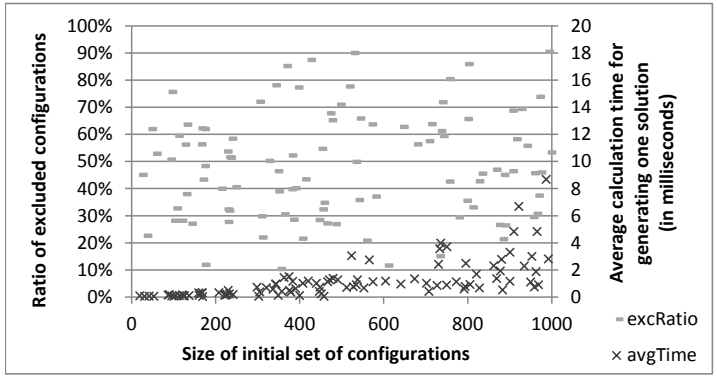
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	38.89%	7.95%	69.72%	0.111	0.012	UNIFORM
Time	2.431	0.03	11.76	2.424	5.875	

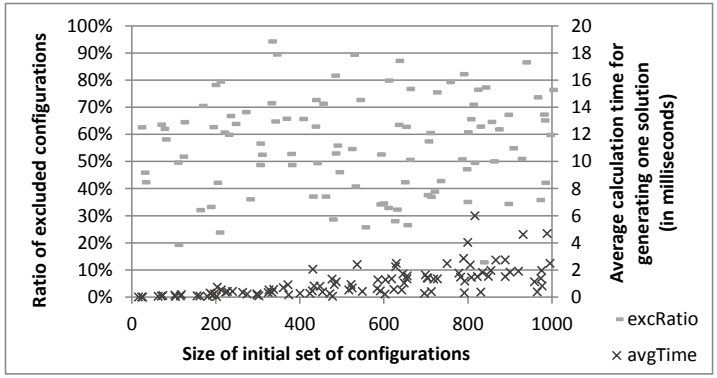
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	57.99%	20.81%	90.41%	0.157	0.025	NORMAL
Time	2.486	0.00	16.54	2.989	8.932	

T3-P1	Cost	Risk	QoS	Complexity
Weighting	0.40	0.20	0.25	0.15
Threshold	100	2	4	0.4





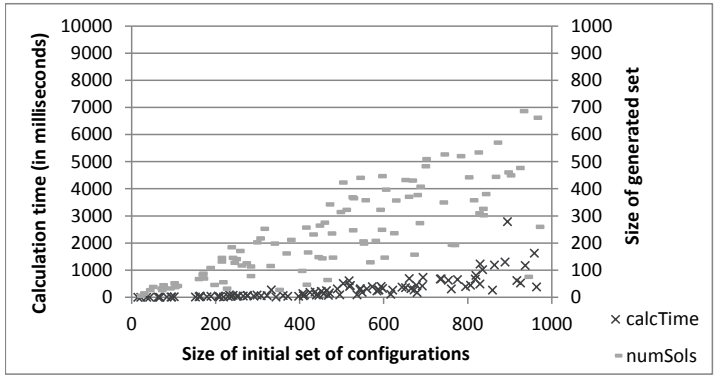


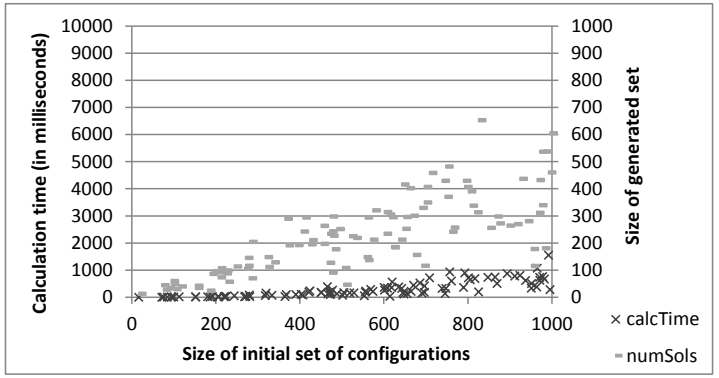


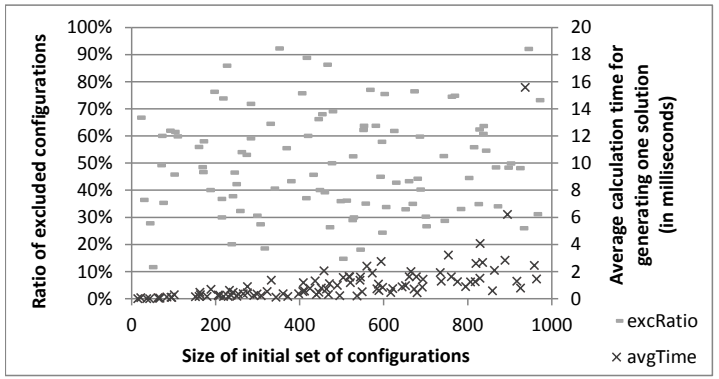
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	48.03%	10.32%	90.37%	0.192	0.037	UNIFORM
Time	1.154	0.03	8.66	1.429	2.043	

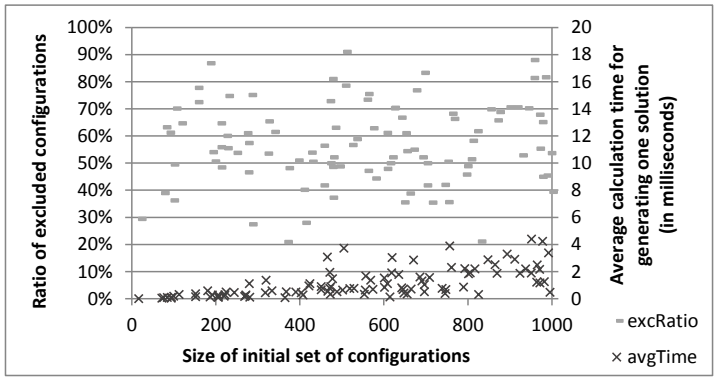
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	55.95%	12.76%	94.19%	0.176	0.031	NORMAL
Time	1.053	0.00	5.99	1.099	1.207	

T3-P2	Cost	Risk	QoS	Complexity
Weighting	0.40	0.20	0.30	0.10
Threshold	100	2	4	0.4





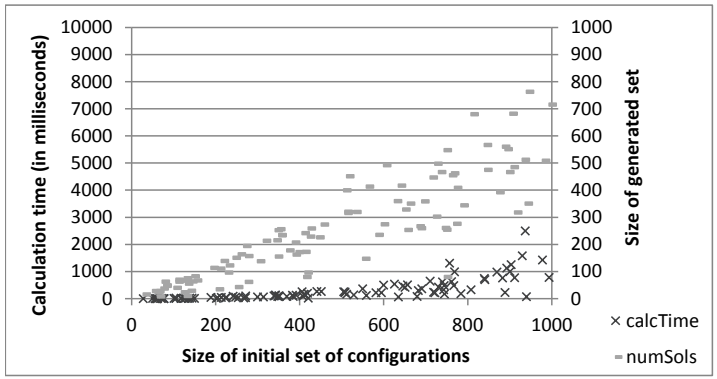


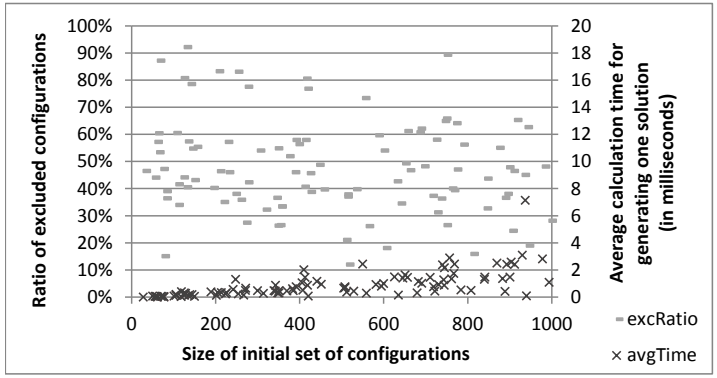


	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	49.44%	11.63%	92.15%	0.184	0.034	UNIFORM
Time	1.101	0.00	15.57	1.754	3.078	

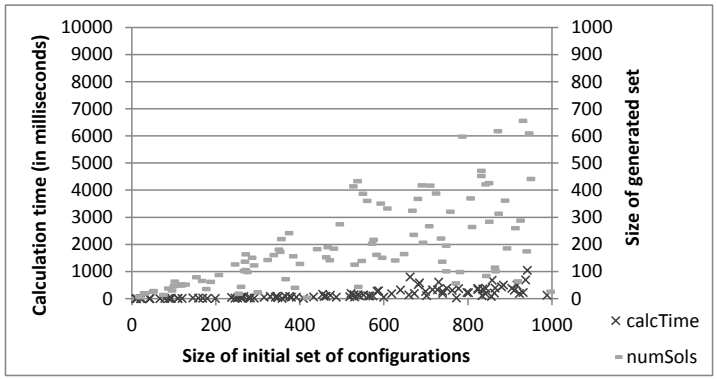
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	56.85%	20.82%	90.89%	0.148	0.022	NORMAL
Time	1.128	0.00	4.38	1.050	1.103	

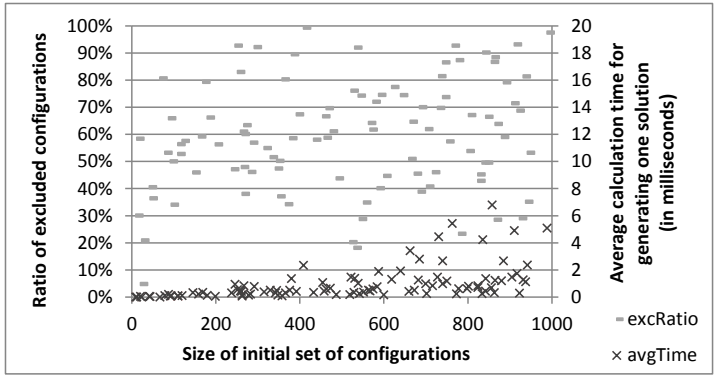
T3-P3	Cost	Risk	QoS	Complexity
Weighting	0.85	0.05	0.05	0.05
Threshold	100	2	4	0.4





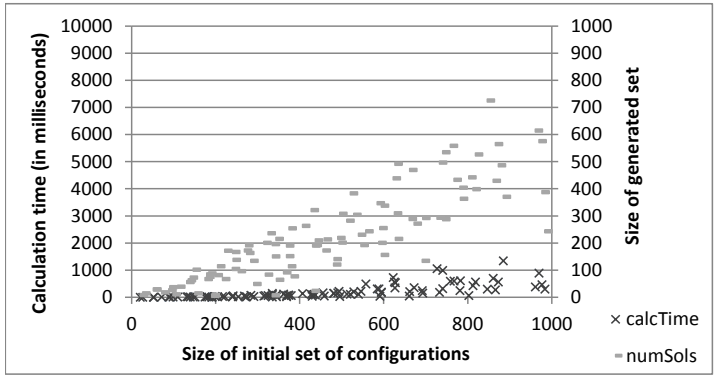
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	47.65%	11.91%	92.06%	0.172	0.030	UNIFORM
Time	0.877	0.00	7.12	1.002	1.004	

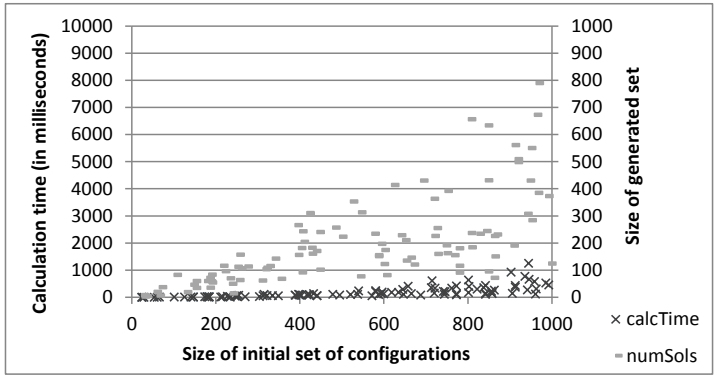


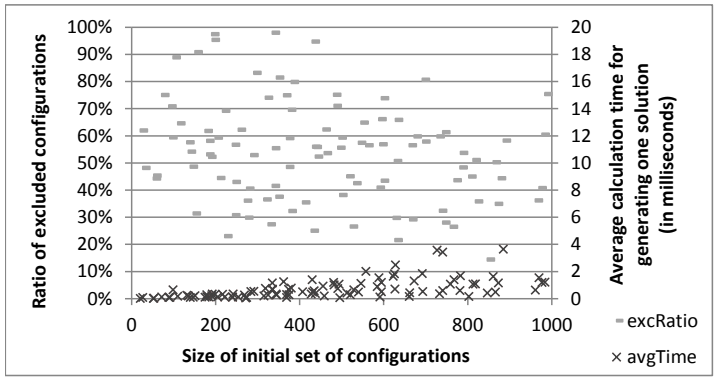


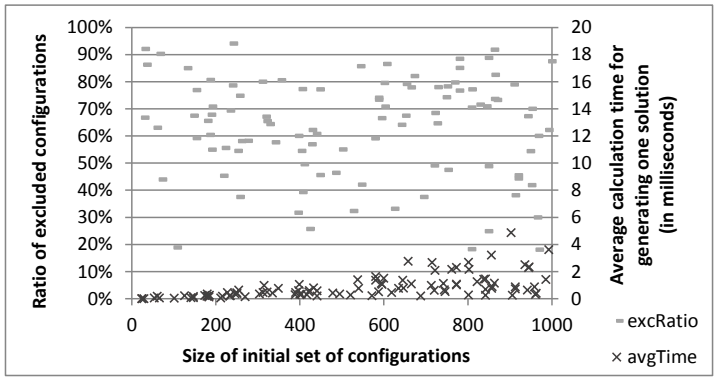
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	59.05%	4.76%	99.27%	0.202	0.041	NORMAL
Time	0.970	0.00	6.78	1.280	1.639	

T3-P4	Cost	Risk	QoS	Complexity
Weighting	0.25	0.25	0.25	0.25
Threshold	100	2	4	0.4





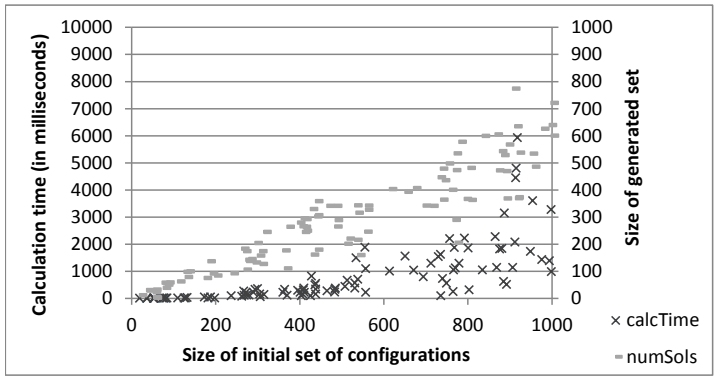


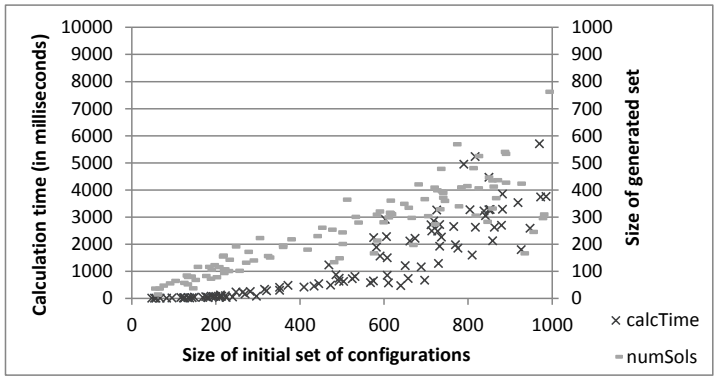


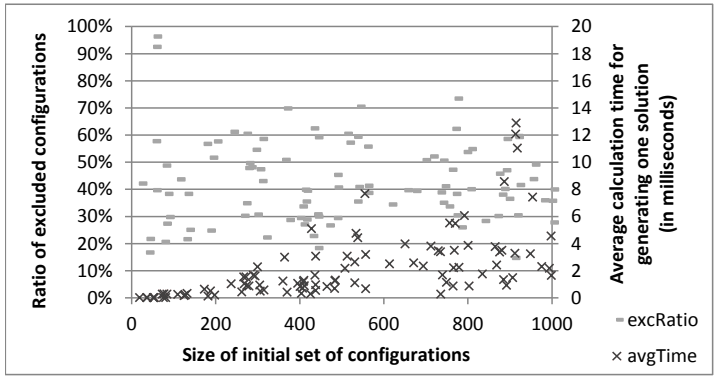
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	53.63%	14.40%	97.91%	0.182	0.033	UNIFORM
Time	0.694	0.00	3.63	0.733	0.538	

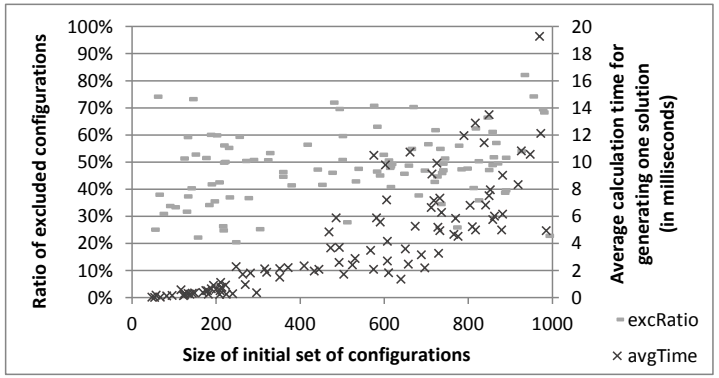
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	63.21%	18.07%	94.04%	0.183	0.033	NORMAL
Time	0.846	0.00	4.86	0.877	0.769	

T4-P1	Cost	Risk	QoS	Complexity
Weighting	0.40	0.20	0.25	0.15
Threshold	50	1	4	0.1





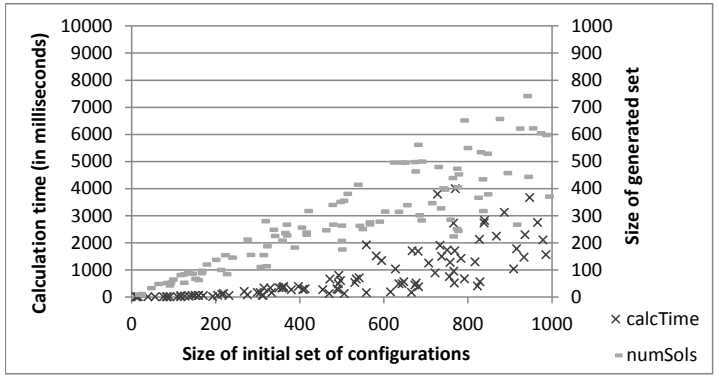


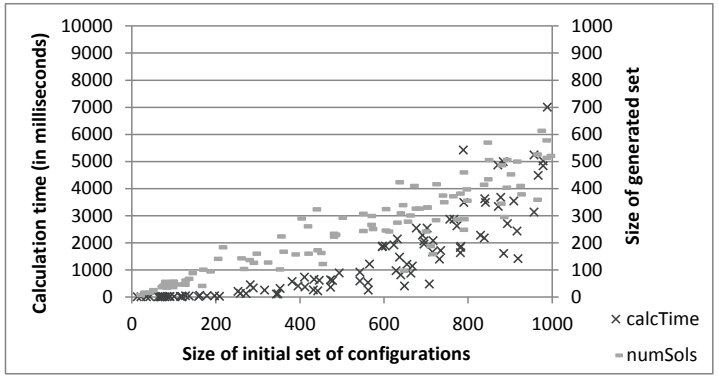


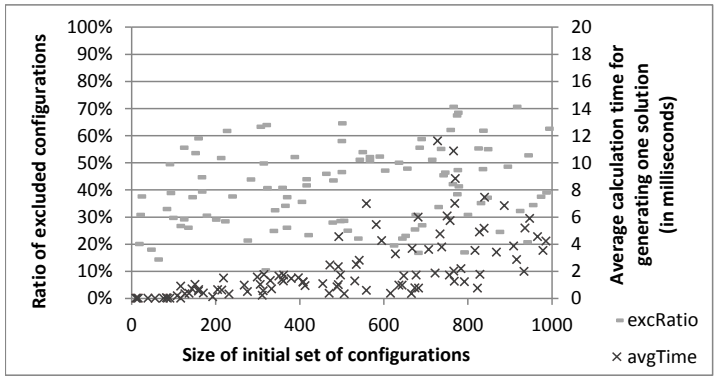
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	42.48%	14.66%	96.30%	0.150	0.023	UNIFORM
Time	2.174	0.00	12.88	2.507	6.284	

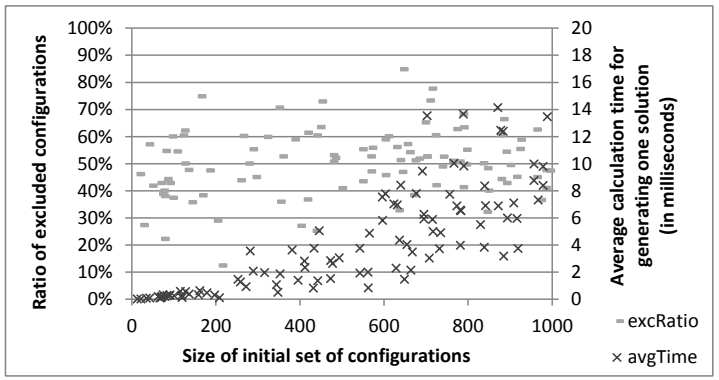
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	48.21%	20.42%	82.09%	0.129	0.017	NORMAL
Time	4.061	0.00	19.27	3.900	15.209	

T4-P2	Cost	Risk	QoS	Complexity
Weighting	0.40	0.20	0.30	0.10
Threshold	50	1	4	0.1









	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	40.70%	10.29%	70.63%	0.144	0.021	UNIFORM
Time	2.267	0.00	11.60	2.417	5.841	

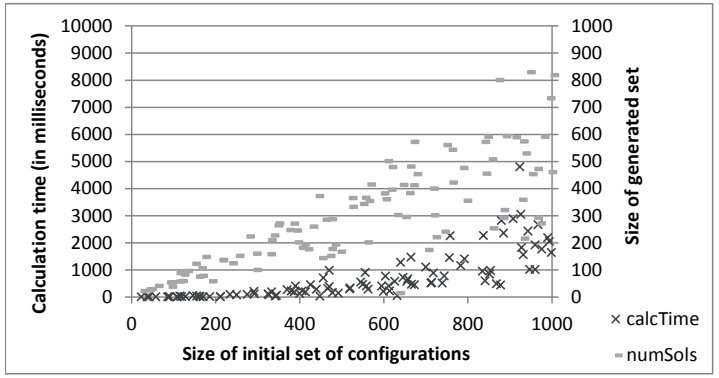
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	50.21%	12.44%	84.84%	0.122	0.015	NORMAL
Time	4.111	0.00	14.13	3.771	14.222	

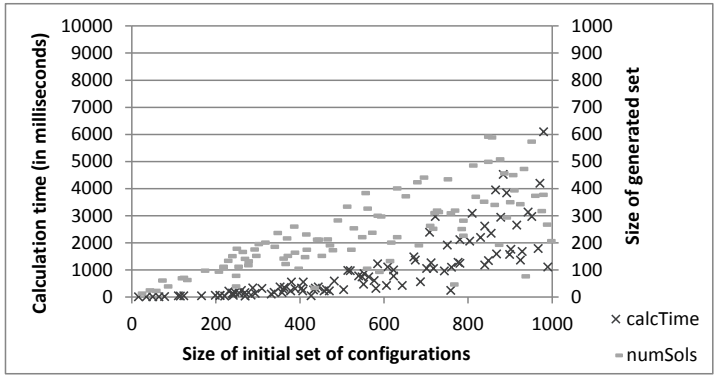
T4-P3	Cost	Risk	QoS	Complexity
Weighting	0.85	0.05	0.05	0.05
Threshold	50	1	4	0.1

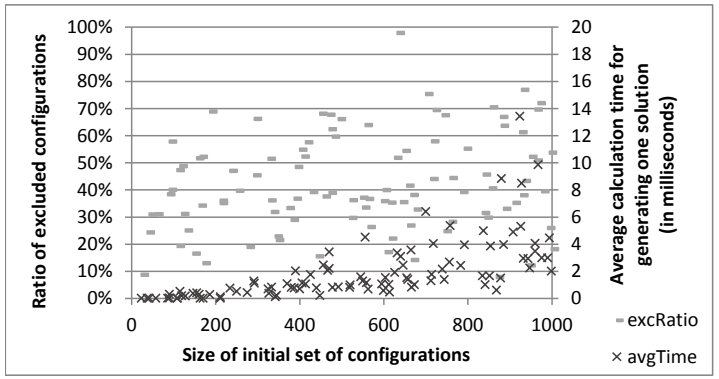
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	41.52%	15.72%	84.43%	0.152	0.023	UNIFORM
Time	2.841	0.00	15.68	3.255	10.597	

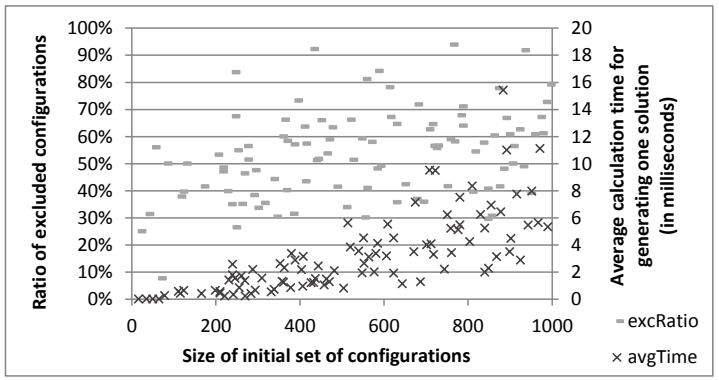
	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	49.71%	20.29%	87.08%	0.140	0.020	NORMAL
Time	3.558	0.00	20.67	3.768	14.197	

T4-P4	Cost	Risk	QoS	Complexity
Weighting	0.25	0.25	0.25	0.25
Threshold	50	1	4	0.1









	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	42.06%	7.83%	97.78%	0.176	0.031	UNIFORM
Time	1.954	0.00	13.42	2.270	5.153	

	MEAN	MIN	MAX	STDEV	VAR	DISTR
Exclusion	53.31%	7.69%	93.94%	0.161	0.026	NORMAL
Time	3.755	0.00	22.81	4.273	18.262	

Abstract

Nowadays service ecosystems rely on dynamic software service chains that span over multiple organisations and providers. They provide an agile support for business applications, governments of end-users. This trend is reinforced by the Cloud based economy that allows sharing of costs and resources. However, the lack of trust in such cloud environments, that involve higher security requirements, is often seen as a braking force to the development of such services.

The objective of this thesis is to study the concepts of service orchestration and trust in the context of the Cloud. It proposes an approach which supports a trust model in order to allow the orchestration of trusted business process components on the cloud.

The contribution is threefold and consists in a method, a model and a framework. The method categorizes techniques to transform an existing business process into a risk-aware process model that takes into account security risks related to cloud environments. The model formalizes the relations and the responsibilities between the different actors of the cloud. This allows to identify the different information required to assess and quantify security risks in cloud environments. The framework is a comprehensive approach that decomposes a business process into fragments that can automatically be deployed on multiple clouds. The framework also integrates a selection algorithm that combines security information with other quality of service criteria to generate an optimized configuration.

Finally, the work is implemented in order to validate the approach. The framework is implemented in a tool. The security assessment model is also applied over an access control model. The last part presents the results of the implementation of our work on a real world use case.

Keywords: Business Process Management, Cloud Computing, Security Risk Management

Résumé

L'essor du Cloud Computing, permettant de partager les coûts et les ressources au travers de la virtualisation, présage une interconnexion dynamique et flexible entre entreprises et fournisseurs. Cependant, cette mise en commun de ressources, données et savoir-faire implique de nouvelles exigences en termes de sécurité. En effet, le manque de confiance dans les structures du Cloud est souvent vu comme un frein au développement de tels services.

L'objectif de cette thèse est d'étudier les concepts d'orchestration de services, de confiance et de gestion des risques dans le contexte du Cloud. La contribution principale est un framework permettant de déployer des processus métiers dans un environnement Cloud, en limitant les risques de sécurité liés à ce contexte.

La contribution peut être séparée en trois partie distinctes qui prennent la forme d'une méthode, d'un modèle et d'un framework. La méthode catégorise des techniques pour transformer un processus métier existant en un modèle sensibilisé (ou averti) qui prend en compte les risques de sécurité spécifiques aux environnements Cloud. Le modèle formalise les relations et les responsabilités entre les différents acteurs du Cloud. Ce qui permet d'identifier les différentes informations requises pour évaluer et quantifier les risques de sécurité des environnements Cloud. Le framework est une approche complète de décomposition de processus en fragments qui peuvent être automatiquement déployés sur plusieurs Clouds. Ce framework intègre également un algorithme de sélection qui combine les information de sécurité avec d'autres critère de qualité de service pour générer des configuration optimisées.

Finalement, les travaux sont implémentés pour démontrer la validité de l'approche. Le framework est implémenté dans un outil. Le modèle d'évaluation des risques de sécurité Cloud est également appliqué dans un contexte de contrôle d'accès. La dernière partie présente les résultats de l'implémentation de nos travaux sur un cas d'utilisation réel.

Mots-clés: Gestion des Processus Métiers, Cloud Computing, Gestion des Risques de Sécurité