Turun yliopisto
University of Turku

# RISKS IN CLOUD COMPUTING

Master´s Thesis
in Information Systems Science

Author:
Anssi Nurmilahti

Supervisor:
Ph.D. Jonna Järveläinen

7.9.2016
Turku

## Table of contents

1 INTRODUCTION ..... 5

1.1 Background ..... 5
1.2 Research questions ..... 7
1.3 Structure of the research ..... 7

2 RESEARCH METHODS ..... 9

2.1 Background for the literature review ..... 9
2.2 Searching for the literature ..... 10
2.3 Assessing the quality of the literature ..... 12

3 CLOUD COMPUTING ..... 13

3.1 Background ..... 13
3.2 The definition of cloud computing ..... 16
3.3 Cloud service models ..... 17
3.3.1 Infrastructure as a service ..... 20
3.3.2 Platform as a service ..... 21
3.3.3 Software as a service ..... 22
3.4 Cloud deployment models ..... 24
3.4.1 Private cloud ..... 25
3.4.2 Community cloud ..... 26
3.4.3 Public cloud ..... 27
3.4.4 Hybrid cloud ..... 28

4 RISKS IN CLOUD COMPUTING ..... 29

4.1 The definition of risk ..... 29
4.2 Risks to data security and privacy ..... 33
4.2.1 Data confidentiality ..... 33
4.2.2 Data integrity ..... 35
4.2.3 Data availability ..... 35
4.2.4 Data location ..... 37
4.2.5 Data privacy ..... 38
4.3 Technological security risks ..... 41
4.3.1 Virtualization risks ..... 41
4.3.2 Risks in web technologies ..... 43
4.3.3 Risks in identity and access management ..... 45
4.4 Relationship risks ..... 47
4.4.1 Lack of trust ..... 47

4.4.2 Lock-in ........ 48
4.4.3 Outsourcing risks ........ 50

5 RISKS IN CLOUD SERVICE MODELS ........ 52

5.1 Risks in IaaS ........ 52
5.2 Risks in PaaS ........ 53
5.3 Risks in SaaS ........ 53

6 MANAGING THE RISKS IN CLOUD COMPUTING ........ 55

6.1 Service level agreements ........ 55
6.2 Standards, frameworks & security models ........ 59
6.3 Trust ........ 62

7 CONCLUSIONS ........ 66

7.1 Conclusions ........ 66
7.2 Limitations and implications for future research ........ 67

REFERENCES ........ 68

## List of figures

Figure 1 The layers in the cloud stack (Youseff, Butrico & Da Silva 2008, 4) ........ 18

Figure 2 The contributors to and effects of vulnerabilities and threats (Grobauer, Walloschek & Stöcker 2011, 51) ........ 30

Figure 3 The complexity of the cloud security (Subashini, Kavitha 2011, 2) ........ 32

Figure 4 Data life cycle (Yu, Wen, 2010, 1) ........ 39

# 1 INTRODUCTION

## 1.1 Background

Cloud computing is an emerging and rapidly growing computing paradigm. According to Columbus (2016, 2–3), the worldwide cloud computing market was $110 billion in 2015, with growth rates of 51% in public IaaS and PaaS offerings and 45% in private and hybrid offerings in the same service categories. Big companies are leading the way; Amazon Web Services (AWS) generated $7,88 billion in revenue – in the Q4 of 2015 alone (Columbus 2016, 5). These numbers, in a class of their own, make it very clear that cloud computing is a serious transformation in the area of computing services.

The reason for these staggering growth figures, future projections and the popularity of cloud computing lies in the way it has made large scale computing easily available. The barriers to entry are significantly lower; even non-existent, and even the smallest players can acquire computing resources previously unavailable to them. There is no longer a need to make significant capital investments in an infrastructure upfront and to hire staff to operate the infrastructure, freeing the resources to focus on the core business. This levels the playing field, enabling the small companies to participate in competition with bigger corporations.

According to Fox, Griffith, Joseph, Katz, Konwinski, Lee, Patterson, Rabkin, Stoica & Zaharia (2009, 7–8), reasons for the fast growth of cloud computing are the rapid rise of mobile applications, introduction of many data-intensive services and the wide adoption of business analytics. Fox et al. (2009, 1) have defined three characteristics exclusive to cloud computing: the cloud users have practically infinite resources at their disposal, there are no strong financial commitments required from the cloud users, and they only pay for the services they actually use. These properties give the cloud the unique financial and functional flexibility that cannot be reached by relying solely on the traditional data center architecture, where the infrastructure is fully owned by the company using it. Different cloud service models combined with different cloud deployment models serve different purposes. These models make cloud computing a relevant alternative to all kinds of business requirements, contributing to its wide adoption in business use.

Despite cloud computing being around for more than a decade, it was only in May 2013, when the United States Federal Government approved the use of cloud computing in federal agencies. The approval was given to the use of Amazon Web Services in federal service provision. Before this approval Amazon had to go through a thorough risk assessment, called FedRAMP. The approval is valid for three years at a time and applies only to data centers that are physically located in the US (Barr 2013, 1).

This careful risk consideration reflects well the uncertainties related to cloud computing. As in any agreement between parties, be it a manufacturing subcontract or an agreement of consultancy services provided to improve a company's business processes, there are risks involved. Due to the complexity of the cloud computing paradigm, also the risks are complex and their management requires broad understanding of the technological and operational aspects of the cloud.

Private companies are generally fast to adapt to new environments, explore new opportunities and create new business models based on new technologies, but the same uncertainties that the authorities and the general public are concerned about, apply also to them. The fact that cloud computing is still under such careful consideration and scrutiny by government authorities is a clear indicator of the many uncertainties associated with it. The widespread news and the ensued debate about the U.S. National Security Agency tapping into the personal data of millions of cloud service customers is hard evidence that there is a need for careful risk consideration.

The ISO 31000 standard defines risk as the effect of uncertainty on objectives, the effect being the deviation from the expected, either positive or negative (Purdy 2010, 882). There are risks that are more probable and predictable than others, as there are risks that are highly unlikely and very hard to predict. Risks also vary greatly in their seriousness, from the smallest financial risks to situations that threaten the reputation, even the whole business of a company. There could be a break-in to the company office building, or a fire in the data center – these two events could have very different outcomes. Some risks can be measured in financial terms. For example, if a company loses one customer due to the realization of a certain risk, it's possible to estimate the costs based on the contract between the company and the customer and by estimating the future business prospects that were lost due to the realization of the risk. But when the reputation of the company is tarnished due to some unlucky event, it's much harder to figure out the costs, as there are no clear, unambiguous indicators for how many potential customers were lost due to the event.

Raz & Hillson (2005, 53–54) note that it's important to differentiate between business risks and operational risks. The business risks are related to financing and insurance, whereas the operational risks are directly related to the business activities of a company, and thus more relevant for cloud computing and this study.

Today, nearly all the information companies handle in order to conduct their business activities is stored and processed electronically using different information systems. In other words, the confidentiality, availability and integrity of the valuable and business critical information is depending on the IT infrastructure of the company. As the technology evolves and becomes more complex, so do the risks, making it an increasingly difficult task to mitigate them. Despite that, a well functioning business should acknowledge these risks in all its operations, and as a result, should have a certain amount of built-in resilience against the risks inherent in the way the business and the core processes are structured. To achieve this resilience, the companies must have a plan for how to deal with the risks. This plan is called a risk management plan, or a risk management framework. The ultimate aim of the planning process is to protect the company's ability to maintain its operations (Stoneburner, Goguen & Feringa 2002, 1).

When companies choose to use the services of cloud computing providers, they simultaneously make the decision to hand over their data to the cloud service provider. This decision introduces a multitude of risks ranging from the physical access to the data, vulnerabilities caused by the complexity of the cloud environment to concerns regarding the security of the data (Subashini, Kavitha 2011, 2-3). A company could also host a private cloud inside its premises using the same technology the big cloud service providers use; however, from the technical viewpoint it would essentially be the same as running a traditional server in-house.

The focus of this study is on the multi-tier cloud computing that consists of one or more parties using the cloud and one or more parties providing the service. These relationships and the amount of parties involved vary depending on the service model and deployment model. More importantly, the focus in the study is limited to those risks that the cloud customer faces, not the risks to the service provider.

## 1.2 Research questions

To get an overview of the concept of cloud computing, the background and motivations behind cloud computing are presented. Thereafter a thorough literature review called a systematic literature review is conducted in order to cover as many of the risks related to cloud computing as possible. The data gathered through the literatures review is then used to further analyze these risks; the aim is to find out which kinds of risks are specifically relevant to cloud computing in general; for each service model and what kind of precautions companies could take in order to effectively mitigate or minimize these risks. The following research questions are answered:

- RQ1: What risks does cloud computing present to companies?
- RQ2: How do these risks manifest themselves in IaaS, PaaS and SaaS?
- RQ3: How can companies manage these risks?

## 1.3 Structure of the research

The study consists of 7 chapters. The second chapter introduces the research method, systematic literature review and clarifies and justifies the methodological and practical decisions made during the review process. The third chapter focuses on the cloud computing paradigm, beginning with the technological background and historical motivations for cloud computing, leading up to the present day, Thereafter the structure of cloud computing is discussed and the service and deployment models are presented. Specific emphasis is placed on the similarities and differences of these models, as they are elemental in assessing the risks in cloud computing. In the fourth chapter, the risks uncovered in the literature review will be presented. The fifth chapter elaborates on these risks, by discussing the relevance of the risks and their relevance to different cloud service models. The sixth chapter is focused on providing different methods or processes for assessing, managing and mitigating the risks in cloud computing. Finally, the seventh chapter summarizes the findings of this study, and states the limitations of the study.

# 2 RESEARCH METHODS

## 2.1 Background for the literature review

The literature review is the basis for any research article. In this paper, the existing literature of cloud computing and the related risks is reviewed. To achieve a wide enough scope and to take all relevant viewpoints into account, a thorough analysis of related literature is an essential requirement. According to Webster & Watson (2002, xv–xvi), a proper literature review should inform the reader of what is the topic of the review, it should aim to cover all viewpoints to the topic and it should be based on multiple qualified sources of information.

Okoli and Schabram (2010, 2) divide literature reviews into three categories: theoretical background reviews, thesis literature reviews and stand-alone literature reviews. The first two types of reviews are concise, in a narrative form and need further empirical data to provide the answers to the research questions. The third type of literature review, the stand-alone review, supports itself without any empirical data; it should be methodologically systematic; it should explicitly state the procedures used in conducting the review, it should aim to include all the relevant material and finally, due to its systematic nature, it should be reproducible (Okoli & Schabram 2010, 1).

Okoli & Schabram (2010, 15) studied the motivations for conducting a systematic review by analyzing various systematic literature reviews and divided their findings in the following motivations:

- To analyze a specific field of study in terms of progress
- To recommend directions for future studies
- To review the applying of a theoretical model in IS literature
- To review the applying of a specific methodology in IS literature
- To produce a model or a framework
- To answer a specific research question.

As cloud computing is still a developing phenomenon, the stand-alone literature review is an apt way of getting a good look into it. According to Torraco (2005, 357–358), the method has it strength in describing new, rising and developing phenomena, as usually such fields do not yet have an extensive body of knowledge to build upon. Since this study is purely theoretical and doesn’t include any empirical data, the emphasis is on the literature review. To ensure sufficient coverage of earlier research, the stand-alone literature review, or later systematic literature review method is applied.

A systematic literature review enables the researcher to find gaps in accumulated research knowledge and identify areas where future research is needed (Webster & Watson 2002, xix). Ideally, the review should contribute to the body of knowledge by giving informed suggestions for future research and so encourage researchers to build on accumulated knowledge in developing new theories (Levy, Ellis 2006, 182).

Levy & Ellis (2006, 182) view the systematic review as a process of collecting, knowing, comprehending, applying, analyzing, synthesizing and evaluating earlier research in order to establish a solid theoretical background for the whole review. Tranfield, Denyer & Smart (2003, 214) divide the systematic review into three different phases: planning, conducting and reporting. Levy & Ellis (2006, 182) use a similar division, they name the phases inputs, processing and outputs, respectively.

## 2.2 Searching for the literature

The review begins by precisely defining the terms of the review, i.e. the purpose, scope and possible limitations of the review; in order to get high quality output, also the input needs to be of high quality (Webster & Watson 2002, xv). However, it's also important that some room for creativity is left; the review will most likely evolve in some way during the review process (Tranfield, Denyer & Smart 2003, 215). The decisions regarding the terms of the review should be properly documented to ensure the reproducibility of the review (Okoli, Schabram 2010, 16–17).

The purpose of this literature review is to get an overview of risks in cloud computing. In the literature search the term risk is considered synonymous with terms and concepts like security issue, vulnerability or threat in order to maximize the yield of the literature search; these terms often appear mixed in various literature (Subashini, Kavitha 2011; Grobauer, Walloschek & Stöcker 2011; Marston, Li, Bandyopadhyay, Zhang & Ghalsasi 2011). All cloud service models and deployment models are in the scope of this review. They are presented in detail in chapter 3.

The next step in the review is the rigorous identification of the search terms that are relevant to the research subject. Levy and Ellis (2006, 190) suggest an initial keyword search to get an overview of what's been written about the subject. Webster & Watson (2002, xvi) suggest doing a backward search, then a forward search. In the backward search, the references of the studies that were found earlier are combed through, and a new search is conducted based on these references. Levy & Ellis (2006, 190–191) further differentiate between a backward references search and a backwards authors search. In the forward search, citations databases are used to find articles that cite the articles uncovered in the original search and the backward search, which will likely increase the richness of the search results. (Webster, Watson 2002, xvi).

The search terms have to be chosen so that they guarantee the comprehensiveness of the search results. This is especially important in the IS field, as the terminology develops rapidly and the use of buzzwords is a regular occurrence (Okoli, Schabram 2010, 7).

According to Levy and Ellis (2006, 190), changes and maturation in the terminology should be taken into account. The initial search for this study was made with search terms such as “cloud computing” and “cloud computing risks”. The initial search helped to narrow down to the following list of search terms:

- Cloud computing risks
- Cloud computing security
- Cloud computing threats
- Cloud computing vulnerabilities
- IaaS risks
- PaaS risks
- SaaS risks

The following library databases were used to conduct the search as they are generally deemed to include high quality articles (Webster & Watson 2002, xvi; Okoli, Schabram 2010, 19–20; Levy, Ellis 2006, 185–186). They also include all of the IS journals that belong to the “Basket of eight”, a group of highly reputable and high quality IS journals.

- ABI/INFORM Global (ProQuest)
- Business Source Complete (EBSCO)
- Emerald Journals (Emerald)
- JSTOR
- SAGE Publications
- ScienceDirect (Elsevier )
- SocINDEX with Full Text (EBSCO)
- Springer LINK
- Wiley Online Library
- Google Scholar

The question is how does one know when the literature search is finished? Levy & Ellis (2006, 192) note that at some point of the review the researcher will most likely start noticing similarity to earlier search results regarding argumentation and methodologies; familiar authors and studies start appearing in the search results more often and no significant new findings are made. That effect was also noticeable in the literature search for this study. Studies that were familiar and already listed in the review protocol started to represent the majority of the search results, i.e. on a result page of 10 articles 6 articles were already accounted for. When most of the search terms started producing similar results the process was deemed finished. A total of 79 Articles were discovered in the search process, 72 of which were included in the final review.

When the pool of material for the actual review is ready, the criteria for the inclusion and exclusion need to be defined. These criteria do not assess the quality of the reviewed articles, only their relevance to the research questions (Okoli, Schabram 2010, 21–22). In order to establish the reproducibility of the study, all the material should be processed according to the same procedures, which is why Tranfield, Denyer & Smart (2003, 216-217) propose the use of forms for documenting the data extraction process.

In this study, material that was at least partially focused on the risks in cloud computing, i.e. the abstract and the contents included mention of risks, threats, security or vulnerabilities, were included in the final review.

## 2.3 Assessing the quality of the literature

Tranfield et al. (2003, 216) and Rousseau, Manning & Denyer (2008, 13–19) present six different criteria for reviewing the literature:

- Construct validity
- Internal validity
- Effect size
- Generalizability
- Intervention compliance
- Contextualization

The first two evaluate the construction and setup of the study, in terms of their explanatory power. The third criterion, effect size, is the relationship between studied variables, it is especially important in meta-analysis and in analyzing quantitative studies. Generalizability examines the claims in regard to their generalizability; whether a phenomenon could be extended to other settings than the one specified in the study. The fifth criterion, intervention compliance, refers to the conditions under which the studied effects were recorded, and whether it is only a very specific set of conditions that produce similar results when applied. Finally, contextualization looks not only at the conditions under which the effects were such as recorded, it also aims at providing information as to why the conditions affect the results in a certain way. Okoli & Schabram (2010, 25–26) suggest on focusing on four aspects in assessing the quality: what does the study claim, how does it back up these claims, how relevant is the provided evidence and finally, is the evidence grounded into practice.

For this study, the quality was assessed based on these four criteria. The claims were analyzed, the studies evaluated according to these claims, and finally, the decision was made regarding the applicability of the study for this review. Of the 72 articles, altogether 69 articles were deemed to be of sufficient quality to be incorporated into the literature review.

# 3 CLOUD COMPUTING

To discuss the risks in cloud computing, it is important to have an understanding about what it is. To accomplish this, the background, development and current state of cloud computing is presented in this chapter, along with some of the technological developments that ultimately made cloud computing possible. The economic rationale behind cloud computing is also briefly discussed, as it is one major reason for the proliferation of the cloud computing and its transformation into the multi-billion industry it is today.

## 3.1 Background

In essence, cloud computing is a form of distributed computing, a computing paradigm where the computing resources are separated from their usage and might be physically dispersed. The early mainframe computers were the first instance of distributed computing. The big mainframes were hosted somewhere away from the users, who would access the mainframes via the terminals that themselves didn't have any computing capabilities. The development of the first computer networks, ARPANET and Multics in the 1960s allowed the remote use of computing resources (Foster et al. 2001, 51). The early visions of cloud computing were recorded around the same time, when MIT professor John McCarthy gave a speech at MIT's centennial celebration in 1961. In his speech McCarthy envisioned that computing one day would be a utility just like the telephone system or the electricity grid, and that the users would pay for the computing resources per actual use. He went even further by stating that the computing utility could become a basis for an entire new industry (Garfinkel 2011, 74).

Grid computing was a big step towards cloud computing. Grid computing is a model that was first introduced in the 1990s, and its idea is to bring together the computing power and storage capacities of several computers over a network using standardized protocols (Foster, Zhao, Raicu, & Lu 2008, 1–2). The result is a computing grid that is similar to a utility like electricity, similar to what McCarthy had envisioned three decades earlier. The focus of grid computing is on the physical storage and computing capabilities, which in the 1990s were the usual bottlenecks in performing tasks that require vast amounts of computing power. Such tasks were becoming increasingly more common at the time in some fields of science, so research institutions and governments were fast to adopt the model. According to Foster et al. (2008, 2), there are three characteristics that can be specifically attributed to grid computing: central control over resources that are not usually coordinated together; standard, open interfaces and protocols; the services provided by the grid are specialized in nature.

While grid computing is effective in providing high computing capabilities for a specific purpose, its computing logic is batch and project oriented, which limits its interactivity (Foster et al. 2008, 3-6). Grid computing is more suitable for one-off projects and ad hoc computing tasks, but less suitable for continuous use in a changing business environment. This limits the applicability of grid computing to a wider range of use, which is why its success has not reached far beyond the borders of the scientific community.

As a simultaneous trend with grid computing, application service providing (ASP) started to emerge. It's a concept of providing software applications to the users via a web interface, without the need to have the application installed on the user's computer (Smith, Rupp 2002, 66). The advantages of the ASP model to companies are the ease of access to costly applications without a big commitment, faster implementation of new applications to business use and lowered overall cost, all while the company could focus on its core business instead of worrying about the technical aspects of the new technology. The ASP paradigm also eliminates the need to deal directly with hardware and software vendors, thus lowering the barrier of adoption. This concerns especially advanced applications (Foster et al. 2001, 55).

Key difference between ASP and cloud computing is that even though both service models are provided over the network, the ASP model is usually based on a more traditional approach where for each customer there's a dedicated server (Krutz, Vines 2010, 39). Thus the biggest advantage of the cloud, scalability, is not present. Another disadvantage of the ASP model is its bulkiness; there is usually little room for customization (Smith, Rupp 2002, 66–67).

The advent of cloud computing in its current form could be attributed to the further development and proliferation of virtualization, a technology that has been around since the end of the 1960's when it was first used on mainframes. By definition, virtualization is the process of creating a virtual representation of a physical computer (Zissis, Lekkas 2012, 584). Virtualization separates the logical from the physical, enabling one physical computer to host multiple virtual computers. This is in contrast to the traditional model where each server is a physical, network-connected device consisting of processors, memory, hard drives and network components (Zissis, Lekkas 2012, 584).

In the traditional model, the provisioning of a new server requires time, money and manpower, invested upfront. A new physical device is required, which might take time to get delivered; thereafter it needs to be installed, configured and launched. All these steps require skilled labor and resources. Virtualization, on the other hand, delivers a far simpler, faster and more cost-effective solution to the same requirement; if a properly configured virtual operating system image is available, the virtual machine can be launched instantaneously, with no upfront commitment and disposed of when it is no longer needed (Zissis, Lekkas 2012, 584). This flexibility is what makes virtualization so prominent: the fast scalability and the ability to quickly provision a large pool of resources with a minimal need for interaction and then discard those resources once no longer needed make virtualization very attractive in comparison to the traditional computing model (Fox et al. 2009, 1). This is especially important for a growing company, as the forecasting of demand for computing is difficult and the infrastructure could become a hindrance to growth. According to Krutz & Vines (2010, 23), some of the main benefits that virtualization provides are:

- Usage-based billing
- Fast scalability
- Economies of scale
- Location independence
- Resilience to hardware failures
- Eased mobility of applications within data centers

Virtualization makes it possible to maximize the utilization rate of computing resources, both for the cloud computing provider and the cloud customer. The customer can quickly dispose of extra capacity, whereas the cloud service provider can maximize the use of resources by hosting multiple customers on the same physical resources (Krutz, Vines 2010, 57). A survey examining 6 corporate data centers revealed that the average resource utilization rate ranged from only 10% to 30% on servers and only 5% on workstations (Marston et al. 2011, 176). According to Fox et al. (2010, 5) the operating costs of large data centers (10000 computers or more) are very low, between 14% and 33% of the operating costs of a medium-sized data center (between 100 and 1000 computers). Cost advantages of this magnitude favor large cloud computing providers such as Google, eBay, Amazon and others. In addition to the financial muscle they have, a critical success factor is that during the dotcom boom in the early 2000's many of those companies already made large-scale investments in suitable hardware infrastructure, scalable database infrastructure and had the knowledge how to efficiently operate such infrastructure (Fox et al. 2009, 5).

When the high resource utilization rate enabled by virtualization is combined with the economies of scale that are gained from operating large data centers, the benefits of large scale virtualized computing come clear: compared to the traditional model, where companies host the servers in-house, cloud computing is capable of providing clear cost benefits. However, the cost advantage is not only beneficial for the cloud provider; it is equally beneficial for the cloud service customers who enjoy lower prices for computing services. At the same time they can focus on their core business, avoid big investments in infrastructure and its costly upkeep and still get the necessary computing services.

These computing models, combined with the rapid development of reliable high-speed Internet connections have paved the way for companies to adopt new models for arranging their business computing needs. The major drivers behind this development are the focus on efficient use of computing resources and the need to quickly adapt to new business requirements and changes (Marston et al. 2011, 177). Fox et al. (2010, 6–7) argue that a shift in web services paradigm was a major driver behind cloud computing. They describe it as a shift from services requiring high levels of commitment and tight relationship between the service provider and service user to relationships where there is little commitment and interaction between the two. Paypal and Google AdSense are good examples, as they enable individuals or entrepreneurs to accept credit card transactions online, without negotiating a contract to do so with a credit card company, or a small web page host to make money with the help of ads on their web page, without having to hire an advertising agency for the job.

## 3.2 The definition of cloud computing

Cloud computing is a widely and loosely used term, and there are conflicting views on what it really is. It is sometimes considered a marketing term used to sell existing technology; that the only true novelty of the cloud is its name (Fox et al. 2009, 3). The reason for the confusion is the nature of the term; it is an umbrella term covering various technologies, service models and approaches to information systems architecture. Vaquero, Rodero-Merino, Caceres & Lindner (2008, 1) address some of that vagueness to the hype surrounding cloud computing.

One of the most widely cited definitions for cloud computing is provided by the U.S. National Institution of Standards and Technology (NIST). They define cloud computing as a computing model that enables omnipresent, on-demand access over a network to a pool of flexible computing resources that can be rapidly taken into use and then abandoned, with little human intervention (Mell, Grance 2011, 2). Foster et al. (2008, 1) define cloud computing as a distributed computing model that effectively utilizes economies of scale in provisioning a pool of virtualized, dynamic and scalable computing power and storage space to provide platforms and services to customers on demand over the network. Fox et al. (2009, 4) have listed three key properties unique for the cloud, from the user's perspective: endless supply of resources at disposal, the gradual addition of resources according to true usage without any initial commitment, and granular billing of the resources used, e.g. by the hour or by the day. Zissis and Lekkas (2012, 584-585) list the key characteristics required of cloud computing as flexibility, scalability, network access, location independence, reliability, the utilizing of economies of scale and sustainability. Vaquero et al. (2008, 3) analyzed various different definitions for cloud computing and concluded: cloud is a pool of accessible, virtual resources that are dynamically configured and re-configured to respond to demand; the use of the resources is charged based on the service level agreements (SLAs). Finally, Buyya, Yeo, Venugopal, Broberg, & Brandic (2009, 4–5) conclude cloud to be a distributed, interconnected system consisting of virtualized computers that are dynamically deployed and presented as a resource to the customer according to the SLA.

Some of these definitions emphasize the ubiquity of the services (Mell, Grance 2011), some place emphasis on virtualization (Foster et al. 2008; Vaquero et al. 2008; Buyya et al. 2009). The pay-per-use model of billing, also called granularity of the service is included in the definitions by Fox et al. (2009) and Vaquero et al. (2008). Zissis & Lekkas (2012) include reliability and sustainability; they argue that these two properties are essential to the cloud although they are the outcome from the other essential properties, mainly the pooling of resources. Apart from these slight variations in the definitions, they share the following elemental properties that define cloud computing:

- Pooled, virtualized resources
- High elasticity and scalability
- Ubiquitous network connectivity and availability
- On-demand, pay-per-use service
- Monitoring and measured service
- Economies of scale, optimal resource utilization

Pooling of resources is made possible by virtualization; which also enables high elasticity and scalability, making it possible to deploy resources when needed and discard them after use. Network connectivity makes the cloud available everywhere; combined with the on-demand, pay-per-use service model the entry barrier for the smallest players is removed, which democratizes the use of computing resources (Mather, Kumaraswamy & Latif 2009, 14). To make the granularity of the services viable, they are monitored and measured; this enables higher resource utilization for the service provider, in turn enabling the economies of scale in the cloud.

These are the key elements that define cloud computing in this study. However, these definitions don’t take a stance to the *structure* of the cloud nor the actual *contents* of the cloud service, as in what kind of services are actually provided. They neither define *the target group* to whom the services are provided; whether they’re available to public or just a limited group of customers. The apparently complex nature of cloud computing requires further categorization, and the next subchapter elaborates on these themes.

## 3.3 Cloud service models

Cloud computing can be classified into different categories, commonly referred to as service models (Mell, Grance 2011, 3; Vaquero et al. 2008, 2). These service models are based on the structure and contents of the cloud; the structure referring to the underlying hardware and software infrastructure, the contents referring to the delivered services. The most widely used classification for cloud services is the SPI-model. The model divides the cloud computing offerings into three different service models (Mell, Grance 2011, 2–3):

- Infrastructure as a Service (IaaS)
- Platform as a Service (PaaS)
- Software as a Service (SaaS)

The service models are built upon a complex structure that is composed of different layers of technology, which are interrelated and dependent on the lower-level layers; combined, these layers form what is called the cloud stack. The lowest level in the stack, infrastructure as a service entails storage, computing and communications; platform as a service adds a development and deployment environment on top of that and finally software as a service delivers a usable application built on top of the infrastructure. One crucial differentiator between these layers in the stack is the level of abstraction they present to the cloud customer (Youseff, Butrico & Da Silva 2008, 1). The lower in the stack the service model is situated, the lower the level of abstraction to the customer. The level of abstraction can thus be used to differentiate and classify the cloud service models. Youseff, Butrico & Da Silva (2008, 3) present a layered model, depicted in Figure 1:

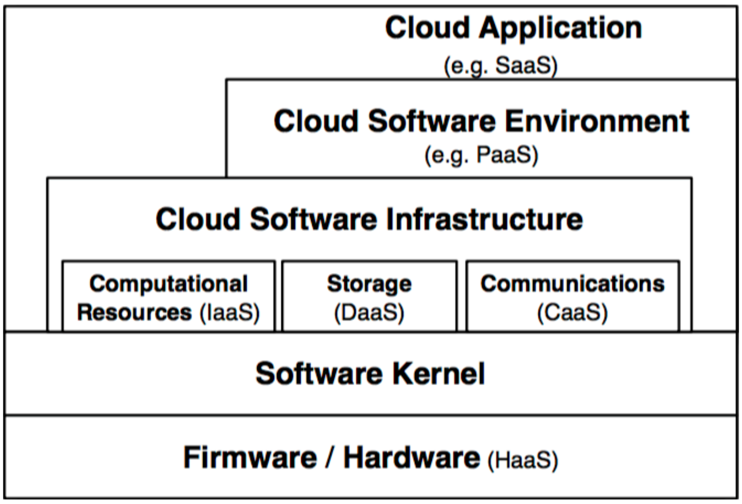
Cloud Application
(e.g. SaaS)
Cloud Software Environment
(e.g. PaaS)
Cloud Software Infrastructure
Computational Resources (IaaS)
Storage (DaaS)
Communications (CaaS)
Software Kernel
Firmware / Hardware (HaaS)

Figure 1 The layers in the cloud stack (Youseff, Butrico & Da Silva 2008, 4)

This representation of the cloud stack extends the SPI-model by slicing the infrastructure layer into further layers. At the bottom of the stack are the hardware and firmware, which are the physical and technological backbone of the cloud. Together they form the most basic service model, hardware as a service (HaaS), which has the smallest degree of abstraction of all service models (Youseff, Butrico & Da Silva 2008, 5–7).

The HaaS model could be utilized by big organizations, which want full control over the resources. The biggest difference between IaaS and HaaS is that IaaS is a multi-tenant model, meaning that the resources are shared between different cloud customers whereas HaaS provides dedicated infrastructure for one single customer (Egwutuoha et al. 2013, 3).

The next layer in the stack, software kernel could be an implementation of a hypervisor or and operating system kernel, depending on what kinds of services are provided. On top of the kernel layer is the cloud software environment, which is divided into computational resources (IaaS), storage (DaaS) and communications (CaaS). This layer provides virtualized resources to the higher-level service models (Youseff, Butrico & Da Silva 2008, 5–6). It could be argued whether the separation of IaaS into IaaS, CaaS and DaaS is necessary, as nearly all definitions for IaaS include processing, storage and networking capabilities (Mell, Grance 2011, 3; Vaquero et al. 2008, 2; Zissis, Lekkas 2012, 584).

The cloud software environment, or PaaS is the next level of abstraction in the cloud stack. This service model provides the cloud customer with an application programming and deployment environment, where they can develop and run applications.

The top layer in the cloud stack is the SaaS level, which is the most sophisticated layer and has the highest level of abstraction (Youseff, Butrico & Da Silva 2008, 3–4).

Lenk, Klems, Nimis, Tai & Sandholm (2009, 2–3) present quite a similar division into service models, which is also based on the SPI-model. They use the same basic division into infrastructure, platform and services, further dividing those components into subcategories such as resource sets, services and environments. They also add an extra layer to the SPI model called HuaaS; Human as a Service. This layer refers to the way some services utilize their user base to further improve the services by way of crowdsourcing. However, as this level mainly refers to consumer-grade services, it can be deemed irrelevant for this study (Lenk et al. 2009, 4).

Armando Fox et al. (2009, 4) have disputed the SPI-model, stating that there is no universal agreement on the contents of the models; instead they argue that the categorization should be made based on the level of abstraction to the cloud service customer and the degree to which the services are managed. Based on the systematic review of the literature, specifically the newer studies, it could well be argued that there is a universal agreement on the SPI-model (Mell, Grance 2011, 3; Zissis, Lekkas 2012, 548). Moreover, as the categorization by Youseff et al. (2008) demonstrated, the SPI model is based on the level of abstraction, contrary to what Fox et al. argue. Although some additional layers and more fine-grained divisions into different service models are presented, e.g. Youseff et al. (2008), the SPI-model model itself is so widely accepted and prevalent in literature that it will be used as a basic categorization in this study.

The service models in the SPI-model all have different purposes and different scope. IaaS has the capability of hosting a vast array of different operating systems and custom applications; the possibilities of PaaS are often limited to the offerings of its own development platform and SaaS is limited to a single application or a collection applications (Fox et al. 2009, 8). Not only are the contents of service models different, also the requirements facing the cloud customer are very different between the models. Different models require different skillsets from the customer; to effectively utilize IaaS or PaaS, more technical knowledge and specific skills are required than using SaaS.

### *3.3.1 Infrastructure as a service*

Infrastructure as a Service can be seen as the most basic form of cloud computing, in the sense that it presents the lowest level of abstraction to the cloud customer. The IaaS-model provides the customer with the basic infrastructure needed to run software, consisting of computing resources, storage space and networking capabilities, which enable the deployment of various applications; including operating systems (Mather et al. 2009, 22). As is imperative to the cloud paradigm and the realization of its promises of flexibility and fast scalability, these infrastructure resources are virtualized. The cloud service provider is in control of the physical infrastructure below the virtualized level (Mather, Kumaraswamy & Latif 2009, 14, 22). This makes the IaaS model much simpler and more flexible to the customer compared to running the servers in-house, as there are less concerns over the technology and its upkeep. It also takes away the need for the long-term resource planning and the related costs (Bhardwaj, Jain & Jain 2010, 62). In some cases the customer might have limited control over some networking resources such as firewalls (Zissis, Lekkas 2012, 584).

Despite the delegation of the physical infrastructure management to the service provider, the IaaS customer is wholly responsible for the software infrastructure, deployment and management of applications, in a similar manner as running a traditional data center (Bhardwaj, Jain & Jain 2010, 62). This is why the IaaS-model requires the most technical the knowledge from the cloud customer. Depending on the choices regarding the operating system and the deployment environment, very specific skillsets might be required to successfully utilize all the potential of the infrastructure.

Lenk et al. (2009, 1–3) divide the lowest-level virtualized resources into a physical resource set and a virtual resource set, which form the basis for the virtual cloud infrastructure. The reason these service sets are divided into physical and virtual is to enable the automated management of both physical and virtual resources. These resource sets enable the automated ramp-up and ramp-down of operating systems, network configurations and other capabilities required to run the virtual infrastructure; they provide an interface between the virtualized computational resources and the higher-level services. Examples of physical resource sets are Emulab and iLO, examples of virtual resource sets are Amazon EC2, Eucalyptus and OpenNebula  (Lenk et al. 2009, 2). On top of these resource sets are the infrastructure services, which are divided into basic infrastructure services and higher infrastructure services. These services form the basis for running cloud applications; they entail the operating system and the application deployment environment (Lenk et al. 2009, 2).

The cloud customer might purchase applications from an external vendor or do some in-house development on their own, but in both cases they are responsible for the running of the applications. One benefit of this model is that the cloud customer doesn't necessarily have to do extensive modifications to their existing software in order deploy them from the cloud, which makes the implementation process of software much quicker (Mather, Kumaraswamy & Latif 2009, 22). Advantages of the IaaS model over the traditional model are its high scalability; the resource redundancy it provides and the separation of data storage and data usage  (Brian et al. 2012, 9).

Amazon EC2 and Simple Queuing Service; Rackspace Mosso Cloud Servers and GoGrid Cloud Storage could be defined as examples of IaaS services. Amazon SimpleDB and S3; 10 Gen MongoDB and Hadoop HBase are examples of database services on the cloud, based on IaaS (Lenk et al. 2009, 4).

### *3.3.2 Platform as a service*

From the technical perspective, platform as a service is a more complex set of services than IaaS. In addition to providing the same basic cloud infrastructure as the IaaS-model, the PaaS-model also provides a development platform for developing and deploying applications. The cloud service provider is responsible for the physical network, servers and the operating system (Zissis, Lekkas 2012, 548).

Mather, Kumaraswamy & Latif (2009, 20) and Krutz & Vines (2010, 40) list the following requirements for PaaS:

- Browser-based development tools
- Integrated development environment to streamline testing and debugging
- Integration with external databases and services
- Inherent support for scalability and reliability, enabled by automation
- Extensive monitoring capabilities, usage-based billing

According to Giessmann et al. (2014, 966) the PaaS development platform is the layer between the IaaS and SaaS; it connects the two layers. It does so by providing a programming environment and an execution environment, and it usually supports a specific programming language, framework and API (Application Programming Interface). The platform usually incorporates a collection of different tools, programming languages and libraries that enable the cloud customer to develop their own applications on the cloud platform, or use available software that has been developed with the programming languages and libraries that the platform supports (Mell, Grance 2011, 2-3).

Compared to web development in general, PaaS greatly increases the number of available and able developers. As developing online applications from scratch requires a great deal of skills ranging from backend development to website configuration, PaaS removes the requirement to have these skills. The platform offers modular blocks of different functionalities, which can be combined to build new services and web applications (Mather et al. 2009, 19). Thus the PaaS developers don’t need to know much about the details of the infrastructure, they only need to familiarize themselves with the API that the platform provides and to be able to work with that (Mather, Kumaraswamy & Latif 2009, 20).

Commonly the platform services are built around the offerings of only one cloud service provider, and due to this restraint cover a vast array of different needs of the cloud customers. This has a potential to create a vendor lock-in, which causes inflexibility and lack of portability of the applications between different runtime environments (Brian et al. 2012, 10). However, some cloud service providers support interoperability, thus making it possible to combine the services of multiple cloud service providers. This enables the cloud customer to use the best possible tools for a given purpose and thus not limiting their choices (Lenk et al. 2009, 5; Mather, Kumaraswamy & Latif 2009 15–17). In addition to the programming language libraries and tools, the platform provides the developers with some common, shared services such as billing, user authentication and authorization. Examples of PaaS are the Google Apps Engine, Amazon Web Services (AWS) and Windows Azure (Mather et al. 2009, 20; Boniface, Nasser, Papay, Phillips, Servin, Yang, Zlatev, Gogouvitis, Katsaros, & Konstanteli 2010, 2).

### *3.3.3 Software as a service*

Technically, SaaS is the most complex service model with the highest level of abstraction to the cloud customer. This makes it also the simplest cloud service model to utilize for the customer: in SaaS a readily usable application is delivered to the customer over the network (Mell, Grance 2011, 2). As in the cloud generally, the cloud applications are accessed through a web interface, or sometimes a client application that is installed on the end-user’s device (Zissis, Lekkas 2012, 584).

Usually the cloud customer is able to do some basic configurations and settings to the software to accommodate their specific requirements, they are also responsible for loading the initial data into the system (Brian et al. 2012, 9). The application itself is hosted on the provider’s infrastructure, and the customer doesn’t have to mind about the infrastructure as it is fully managed by the vendor  (Mather et al. 2009, 18).

The SaaS-model has the potential to significantly lower the software licensing costs and personnel costs of the cloud customer by delegating the ownership and management of the software to the cloud service provider. This is possible as the model separates the possession and the use of the software (Turner, Budgen & Brereton 2003, 39). The cloud service provider also benefits from having their applications running on their own infrastructure, as it significantly eases the oversight of the intellectual property rights; the risk of illegal copying of software is practically non-existent (Mather, Kumaraswamy & Latif 2009, 18).

The term SaaS is sometimes used interchangeably with SOA, or service oriented architecture. This confusion is understandable as the two models are closely related and both enable service delivery over the network. The difference is that SaaS is a model for software delivery whereas SOA is a model for software construction (Laplante, Zhang & Voas 2008, 46–47).

The risk of vendor-lock in, which is also present in IaaS and PaaS to an extent, is significantly higher in SaaS. The root cause is the same, namely proprietary APIs and technologies, which could make it difficult for an organization to extract their own data from the cloud (Fox et al. 2009, 15). Even if the data could successfully be transported out of one service provider’s cloud, it would be in a proprietary form that might not be directly compatible with another service. This could further exacerbate the problem, as building new interfaces for data export and then modifying the data so it could be used elsewhere (ENISA 2009, 26). Another contributing factor to the lock-in problem is the high level of abstraction, which renders the underlying application infrastructure invisible to the cloud customer and thus keeps them in the dark regarding the structure of their data and the operating logic of the cloud applications.

Due to its low barrier of adoption, the amount of available SaaS services has proliferated in recent years. There’s a marketplace containing a very broad range of services, although in some particular kinds of applications the offerings are significantly broader. Benlian, Hess & Buxmann (2009, 366–367) studied the adoption of SaaS services, and found that companies are more likely to utilize a SaaS application for general, non-strategic tasks such as office and collaboration tools, whereas utilizing SaaS for the more strategic business applications such as ERP was uncommon. Common examples of SaaS offerings are the Google Drive and Salesforce CRM offerings (Mather, Kumaraswamy & Latif 2009, 44).

## 3.4 Cloud deployment models

As the cloud service models define *what* services are provided to the cloud customer, the cloud deployment models define *where* and *by whom* the cloud computing services are hosted. This is an important distinction, as the choice of the deployment model can have far-reaching consequences regarding the responsibilities between the cloud service provider and cloud customer; the availability of the services; or even the law that is being applied to the services (Mather et al. 2009, 31–33). The deployment models are functionally and technologically independent of the service models, although some combinations are more common than others; SaaS is usually deployed via the public cloud (Krutz & Vines 2010, 43). The most common way of defining the different deployment models is the division into four different categories based on the degree of privacy and the user base of the cloud (Mell & Grance 2011, 3; Subashini & Kavitha 2011, 2; Zissis &Lekkas 2012, 584):

- Private cloud
- Community cloud
- Public cloud
- Hybrid cloud

Private cloud is exclusive to a certain organization; community cloud is exclusive to a group of organizations. Public cloud is open to all users and hybrid cloud is a combination of private and public cloud. Sometimes these models are further divided into internal cloud and external cloud; the distinction is made based on the location of the cloud infrastructure. Internal cloud is hosted within the perimeter of the organization; external cloud is hosted outside the perimeter (Krutz & Vines 2010, 53). On these dimensions private cloud is defined as internal and public cloud as external; hybrid cloud and community cloud fall somewhere in between (Mather, Kumaraswamy & Latif 2009, 18).

The deployment models have many practical implications to the cloud customers, mainly concerning privacy and security (Subashini, Kavitha 2011, 9). Only the private cloud and community cloud can be wholly trusted regarding the access and consumption of the services, as they are the only cloud deployment models where the cloud customer or customers get a dedicated infrastructure. Hybrid cloud and public cloud are inherently untrusted, as the access to the cloud and the data will always be routed through public infrastructure. This presents the increased risk of the data being routed through a hostile infrastructure, posing a serious threat to the confidentiality and integrity of the data (Subashini, Kavitha 2011, 9). As the cloud computing is based on the applications and data residing in large centralized data centers away from organization's own control, new threats to the security of the company data are introduced compared to the traditional data center architecture (Subashini, Kavitha 2011, 2–3). Many of these threats are related to the data security and technological security of the cloud, which will be analyzed in detail in chapter 4.3.

Depending on the chosen deployment model and the chosen service provider, the company's control over their own data and systems can be greatly reduced. There is also the possibility of the cloud service provider gaining too much leverage if they are the sole service provider or otherwise have extensive control over both the systems and the data, leading to a vendor lock-in (Zhao et al. 2010, 189). The risk of vendor lock-in and other risks related to the relationships between cloud providers and cloud customers will be presented in chapter 4.4.

The deployment model affects the risks in cloud computing to some extent. The magnitude of these effects depends heavily on whether the deployment model is external or internal, as defined by Krutz & Vines (2010, 53). Based on the literature review, however, the risks are more dependent on the service models rather the deployment models. Moreover, as specified in the research question RQ1, the focus in the study is on the risks in different cloud service models, not the deployment models. Due to these limitations, the deployment model -specific risks are shortly presented in combination with the relevant deployment model in the following subchapters, but they are not separately addressed in chapter 4, when cloud-specific risks are presented in more detail.

### *3.4.1 Private cloud*

A private cloud is by definition exclusive for a certain organization. Hosting a private cloud requires significant resources from an organization, as the required technology is expensive and running the infrastructure requires expertise. The private cloud can be hosted by the organization itself, an external service provider or a combination of both. The cloud infrastructure could be physically located at the user's premises, at the service provider's premises or again a combination of both (Mell, Grance 2011, 3; Zissis, Lekkas 2012, 584).

Mather et al. (2009, 24) further divide private cloud into dedicated cloud, community cloud and managed cloud. Dedicated cloud is hosted within the company perimeter and operated by the organization's own staff; the community cloud is hosted on the premises of a third party but owned and operated by a vendor and managed cloud is hosted on the organization's premises but managed by a service provider. In this division the community cloud is defined to be a subclass of private cloud, as it is not open to the public. However, in this study it will be handled as its own, separate deployment model, according to the NIST definition (Mell, Grance 2011, 4) and it will be more thoroughly discussed in the following chapter.

Some of the security issues specifically linked to cloud computing and present in all other deployment models, such as the security issues in virtualization, are absent in the private cloud. When an organization hosts its own cloud on company premises, it faces the same security risks as the traditional infrastructure but no new threats are introduced only because resources are virtualized, as there are no unknown parties using the same infrastructure (Zissis & Lekkas 2012, 585). The oversight and control over the cloud infrastructure are very strong in private clouds, which in turn increases the level of security in the cloud  (Mather et al. 2009, 23–24).

However, these security advantages come at a cost, as the full potential of virtualization cannot be reached. The economies of scale and high resource utilization can only be realized when the resource pool and the user base are large enough to enable the statistical multiplexing, or co-residency on the computing resources in a way that maximizes the utilization rate (Fox et al. 2009, 5). The private cloud is by definition exclusive to one organization, which limits the user base and dictates that the infrastructure be scaled according to the estimated peak demand of that organization. To accommodate peak demand, the resources need to be overprovisioned, which automatically leads to the underutilization of resources during quieter times (Garrison, Kim & Wakefield 2012).

As a remark, it could be disputed whether private cloud even fulfills the definition of cloud computing. It consists of pooled, virtualized resources, but the elasticity and scalability is only limited to the maximum installed capacity; it is network-connected, the resources are provided on-demand and the services are monitored and measured; however, the economies of scale and optimal resource utilization are absent. Further, the private cloud is typically hosted in a traditional data center on company premises (Krutz & Vines 2010, 48). Due to the similarities to in-house infrastructure and the limitations stated in the end of chapter 3.4, the risks in private cloud are even less relevant to this study than the risks in other deployment models. Private cloud poses no unique security risks to organizations compared to the traditional model.

### *3.4.2 Community cloud*

Community cloud is a form of similar to a private cloud regarding exclusivity. Instead of being exclusive to only one organization, it is used by multiple organizations. The motivations for sharing a private cloud infrastructure could stem from the shared needs of the organizations in the community (Mell, Grance 2011, 3). It could be that the companies operate in a field with similar needs for security and redundancy in their IT infrastructure; or that the companies share a need for certain kind of compliance (Zissis, Lekkas 2012, 584).

According to Krutz & Vines (2010, 46) the cloud should fulfill the following criteria in order to be called community cloud:

- Openness
- Community
- Graceful failure
- Convenience and control
- Environmental sustainability

The openness refers to being free from vendor and technology lock-in; community refers to the communal ownership and shared responsibility. Graceful failure is the independence of any single organization; the effects of possible outages in one organization are limited only to that entity and thus don’t hinder the availability of the cloud to others. The convenience and controls refer to the distributed control over the resources; no single community member has excess power over the others. Environmental sustainability is made possible by the concentration of resources to the community cloud, which lessens the need for the community members to run their own infrastructure (Krutz, Vines 2010, 47). The managing of the community cloud could be complex, due to technological complexity presented by the differences in IT infrastructure of the community members.

### *3.4.3 Public cloud*

Public cloud is the most widely adopted form of cloud computing, usually when cloud computing is discussed the subject of that discussion is the public cloud. By definition, public cloud offers powerful, highly scalable computing resources to the public; or individuals or organizations who need computing services (Dillon, Wu & Chang 2010, 28). Public cloud is the most common of the cloud deployment models, and it’s the most common platform for providing SaaS. The two models are a natural fit, as SaaS is by nature suited to applications, which have a low barrier of adoption and a low level of commitment (Fox et al. 2009, 7). The popularity of the public cloud could directly be attributed to the prevalence of SaaS cloud offerings. Examples of public cloud are Amazon Web Services, Microsoft Azure and Google App Engine (Krutz, Vines 2010, 45). The responsibilities of different parties are clear in the public cloud: the service provider has full ownership and control over the infrastructure, deployment environment and applications (Dillon, Wu & Chang 2010, 28).

Public cloud is the deployment model that presents the most risks. Based on the division by Krutz & Vines (2010, 53), it is the only deployment model that could strictly be defined external, meaning that the physical infrastructure lies wholly outside the organization’s safety perimeter. To add to this, multiple customers are using the same infrastructure, which causes problems regarding data security, locality, integrity and segregation; network security; authorization and authentication. The various security risks regarding virtualization also need to be taken into account (Subashini, Kavitha 2011, 4). These risks will be more thoroughly discussed in chapter 4, based on the relevance of the risks to different service models (RQ1).

### *3.4.4 Hybrid cloud*

Hybrid cloud is a combination of the previous cloud deployment models. It consists of at least two of them, but could also include all three (Krutz, Vines 2010, 49). The private and public infrastructures that form the hybrid cloud are functional entities as such, but they are interlinked so that the applications on either infrastructure can connect to each other and thus enable load balancing and application portability (Mell, Grance 2011, 3; Zissis, Lekkas 2012, 584). Usually organizations employ the isolated, private infrastructure to host the business-critical data and run their core infrastructure and applications, whereas the less critical applications can be deployed in the public cloud (Mather et al. 2009, 25).

Hybrid cloud enables the rapid deployment of computing reserves from the cloud if there's rise in demand. During normal levels of usage, the private cloud infrastructure is able to handle the load. However, in case of a sudden demand surge, the hybrid cloud's ability to transfer a part of the load over to the public cloud enables the customer to handle the peak load without acquiring extra capacity. They can just provision the right amount of virtual servers from the public domain to perform an exceptionally heavy task and then release the servers back into the resource pool when the task is finished (Krutz & Vines 2010, 49). This inter-cloud load-balancing functionality is called cloud-bursting, and it ensures the company only pays for the amount of computing power they actually use, thus minimizing the redundancy (Zissis, Lekkas 2012, 584).

With the focus on the essential properties of the cloud, different cloud service models and deployment models, the aim of this chapter was to provide an understanding of what cloud computing is, how it can be distinguished from other forms of web services or hosted computing and how the cloud computing services are constructed. In the light of this understanding, the risks that are present in cloud computing can be put into the appropriate context and given the appropriate weight.

# 4 RISKS IN CLOUD COMPUTING

## 4.1 The definition of risk

This chapter aims to present different risks in cloud computing, each risk being part of the answer to the RQ1. In order to discuss these risks, it is first imperative to properly define what is meant by risk; what limitations the term possibly has and what other closely related terms are commonly used. The ISO 31000 -standard defines risk as the effect of uncertainty on objectives (Purdy 2010, 882). The objectives are the goals set by an organization; uncertainty is the effect of external and internal forces that the organization is not in total control of. This definition of risk simplifies the risk management to an optimization process that enhances the likelihood of the organization of reaching its goals (Purdy 2010, 882).

Kaplan & Garrick (1981, 12) present a twofold definition for risk. The first component, uncertainty + damage depicts the uncertainty related to risks; that they can never be accurately predicted and they by definition include a potential of harm or damage. The second component of risk evaluates the relationship between a potential hazard and the safeguards in place to prevent that hazard from realizing; the risk magnitude is directly related to both. Stoneburner, Goguen & Feringa (2002, 8) and Grobauer, Walloschek & Stöcker (2011, 50) define risk as a function of a potential threat utilizing a vulnerability and the resulting negative impact on the organization. In all of these definitions, risk is a composition of different elements. There is uncertainty and damage; there are threat or risk sources; vulnerabilities and threats. To properly discuss risks in cloud computing, it's important to understand what these elements are and how they interact with each other.

Stoneburner, Goguen & Feringa (2002, 15) define vulnerability as a weakness in system design, implementation, security procedures or internal controls. According to Grobauer et al. (2011, 50), vulnerability is the likelihood that an asset cannot resist the actions by a threat agent. When a threat agent uses such force that exceeds the object's capability to resist that force, the object is vulnerable. Threat is then the potential of a threat source to exploit, either accidentally or with intent, a specific vulnerability (Stoneburner, Goguen & Feringa 2002, 12). Stoneburner et al. (2002, 13) define threat-source as a circumstance or event that has the potential to cause harm to information systems. The ISO 31000-standard defines risk source as an element that has intrinsic potential to increase risk (Leitch 2010, 888). Grobauer et al. (2011, 51) argue that the damage caused by the realization of threats doesn't differ at all in cloud computing compared to the traditional infrastructure. The loss of customer data has similar negative consequences to the organization regardless of their chosen method of providing their IT infrastructure.

A detailed view of risk is presented by Grobauer et al. (2011, 51). It takes into account the risk sources on the left hand side and the potential damage caused by these risks on the right hand side in Figure 2.

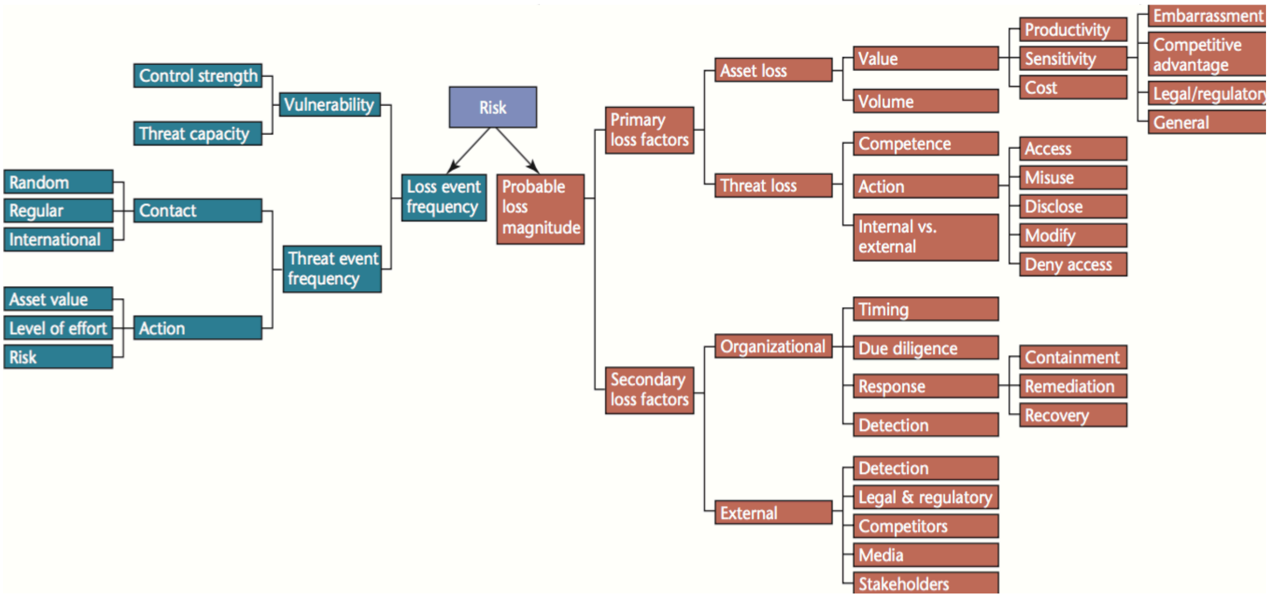

Figure 2 The contributors to and effects of vulnerabilities and threats (Grobauer, Walloschek & Stöcker 2011, 51)

The loss event frequency is a function of vulnerabilities and threat event frequency. Vulnerabilities are affected by the magnitude of the threat they pose to the organization and the controls in place to minimize their effect. The threat event frequency depicts how often there are attempts to exploit existing vulnerabilities. This frequency is affected by a combination of the attackers' motivations and their potential gain from the attack. It's important to note that threat events cannot be accurately predicted or controlled whereas vulnerabilities can be treated or their effects minimized (Grobauer, Walloschek & Stöcker 2011, 51–52). Based on these definitions, risk is the outcome of multiple factors. Risks are caused by known and unknown vulnerabilities and their effects on the organization but risks can also emerge from the relationship between the service provider and cloud customer. In other words, a risk can also exist independent of vulnerabilities.

Grobauer et al. (2011, 52) debate vulnerabilities that are specific to cloud computing; they argue that for a vulnerability to be cloud-specific it needs to:

- Be an inherent part of the technology that enables cloud computing
- Have its cause in one of the essential characteristics of the cloud

- Have its cause in the incompatibility between established security controls and cloud computing
- Be universal in modern cloud computing offerings

To exemplify, when virtualization is utilized within the organization's safety perimeter to deploy virtualized resources, it doesn't pose any new risks compared to the existing infrastructure, as the safeguards in place to protect the traditional IT infrastructure also protect the virtual resources. However, when the virtual resources are deployed on a public, shared infrastructure, multitenancy introduces vulnerabilities. VM hopping is the event of gaining access from one VM to another VM running on the same host machine; VM mobility is the event of unauthorized copying of the data on VM disks over the network (Tsai, Siebenhaar, Miede, Huang, & Steinmetz 2012, 34–35). Another example of cloud-specific vulnerability is the availability of the cloud services. The cloud customer has little control over an event or incident that affects the service provider's infrastructure. Thus, they rely completely on the service provider to shield the cloud from denial of service (DoS) attacks (Subashini, Kavitha 2011, 7). A third example is data lock-in in SaaS; the proprietary cloud applications might pose challenges for exporting the customer data out of the cloud, thus increasing the vendor's leverage and increasing the risk of data lock-in (Brender, Markov 2013, 729).

As the technology behind cloud computing is complex, so are the risks. SaaS presents different security challenges to the cloud customer than IaaS simply for the reason that in SaaS the customer has far less control over their data; they might not even know where their data is physically stored or which other companies use the same physical storage (Kaufman 2009, 62). As another example, the concern of data lock-in is present in all service models but much more relevant to SaaS than for IaaS. The cloud customer has far less control over the infrastructure and their data in SaaS as they would have in IaaS, for example (Jansen, Grance 2011, 3–4). A third example is the use of third-party applications and services. It is possible in PaaS, but in such cases part of the responsibility for securing the services is shifted to the third party, which adds complexity to the cloud infrastructure. The cloud service provider might also be unwilling to expose details of the security controls in their cloud due to the possibility of the information leaking to the third a party with malicious intent. This further reduces transparency and trust in cloud (Mather, Kumaraswamy & Latif 2009, 56). The complexity of cloud technology and the risks that it poses are depicted in Figure 3, which shows the cloud deployment models, service models and the essential properties of the cloud as a stack, along with different risk sources.

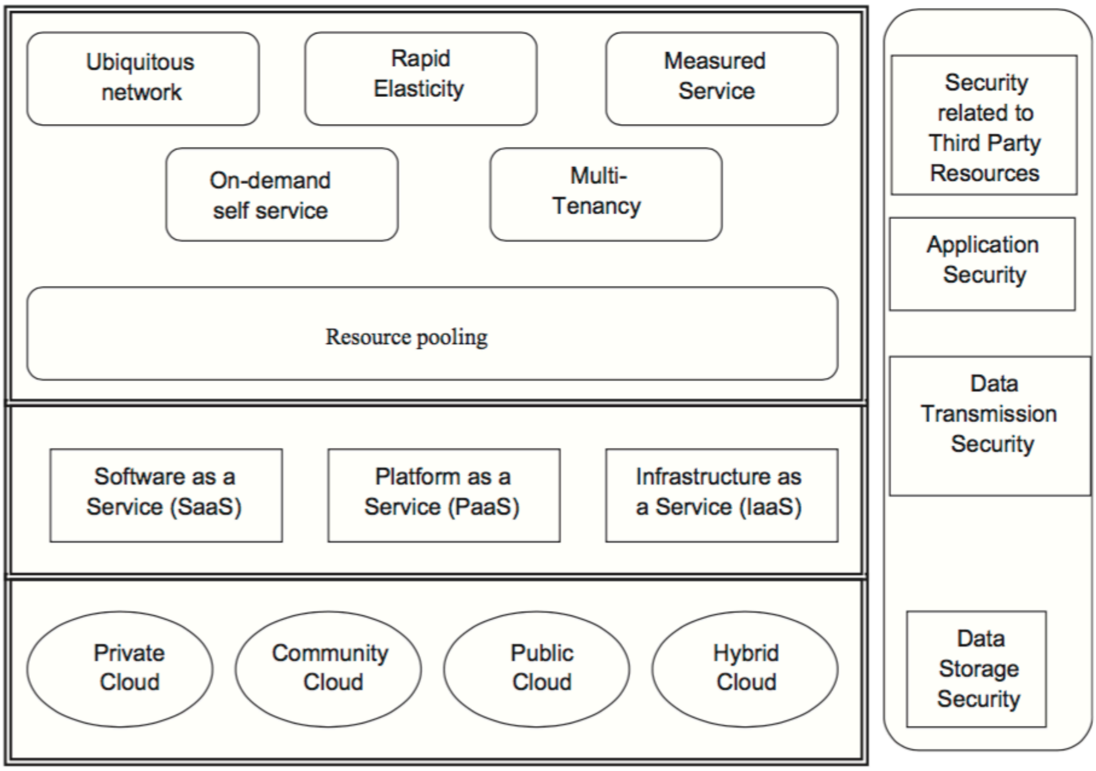

Figure 3 The complexity of the cloud security (Subashini, Kavitha 2011, 2)

On the right hand side different security concerns are presented, aligned with the parts of the cloud that are relevant to each security concern. The cloud deployment models have an impact on data security, whereas the service models have more relevance to data transmission and application security, the essential characteristics of the cloud pose (Subashini, Kavitha 2011, 5, 7). Naturally, this is only a simplified picture of the risks in cloud computing; nonetheless, it gives a good idea of all the factors in play when assessing the risks in cloud computing.

Vaquero, Rodero-Merino & Morán (2011, 18) argue that many of the concerns related to cloud computing are in fact well known risks that have prevailed for long in web services and data hosting services, and are not in any way exclusive to the cloud. Rather, it is how cloud computing combines mature technologies with new, innovative approaches to provide services to customers what makes the risk landscape of the cloud unique.

In the following subchapters various risks and security concerns in cloud computing are discussed. These risks and concerns are grouped thematically, and the subchapters take no stance to the relevance of the particular security concern or risk to the cloud service model or deployment model. The relationship between the risks and the service models, as defined in the RQ2, will be discussed in detail in chapter 5. The deployment model has implications for the risk on a general level, in the sense that private cloud has as high a degree of security as the organization's safety perimeter and safety measures allow, i.e. the risk level is the same as it is for in-house infrastructure run by the same organization (Mather, Kumaraswamy & Latif 2009, 23). Also, as discussed in chapter 3.4.1, the risks of private cloud and traditional infrastructure don't differ much from each other.

Public cloud, on the other hand, has more security concerns, as it faces the public Internet and thus provides much more attack surface and more security risks; this notion shifts the emphasis of the risk analysis onto the public-facing deployment models. This justifies the discussion of the risks in cloud under the assumption that the deployment model is a public one, i.e. public cloud, community cloud or hybrid cloud. The aim in the following subchapters is to provide an answer to RQ1.

## 4.2 Risks to data security and privacy

Perhaps the most serious concern companies face when moving their services to the cloud is the security of their data (Fox et al. 2009, 15–16). As the data crosses the traditional enterprise boundaries, it is no longer subject to the company's physical security measures such as control of physical access to company locations, or the technical security measures protecting the information systems (Subashini, Kavitha 2011, 4). According to Kaufman (2009, 62), the cloud obfuscates not only the location of the data, but also the co-tenants, or other organizations, which store data on the same physical storage. To ensure information systems security, the threats need to be identified and relevant measures taken to prevent the risks caused by the threats from realizing. According to Von Solms & Van Niekerk (2013, 2) information security is the process of ensuring the confidentiality, integrity and availability of information. Data security is a subset of information security, so the same requirements apply. A very common way of categorizing data security is the division into confidentiality, integrity and availability of data (Von Solms, Van Niekerk 2013, 2).

### *4.2.1 Data confidentiality*

The first element, confidentiality, means that only such parties and systems, which have specifically been granted access to confidential data, can access that data (Zissis, Lekkas 2012, 586). The cloud presents additional challenges to data confidentiality. The openness of the public cloud means there are multiple users for the same resources, thus increasing the risk of unauthorized access. Due to the sheer volume of data and transactions, also the security measures need to be capable of handling much more data than in a traditional infrastructure. There could also be confusion regarding responsibilities if the cloud is hosted by multiple organizations; this could hurt the ability to quickly react to security threats. Finally, as virtualization obfuscates the resources and the infrastructure boundaries, it is difficult to identify and then isolate a resource in case of a data breach (Chen & Zhao 2012, 648).

The confidentiality of data is also subject to legal considerations. This is especially true for public cloud services and sensitive data that are strictly regulated, such as financial information or health data. An example of such regulation is the PCI DSS (Payment Card Industry Data Security Standard), which is a set of requirements regarding information storage and auditability (Subashini, Kavitha 2011, 7). Another example is the Health Insurance Portability and Accountability Act (HIPAA), which regulates the access to and handling of health information in the US (Fox et al. 2009, 15).

The high degree of automation in the cloud combined with the complexity of the layered infrastructure introduces technical vulnerabilities such as interception of data in transit and session highjacking, which could lead to privacy breaches (Zissis, Lekkas 2012, 587). In general, the complexity greatly increases the attack surface in the cloud, and the cloud access technologies such as SOAP (Simple Object Access Protocol) or REST (Representational State Transfer) present additional vulnerabilities (Almorsy, Grundy & Müller 2010, 5). These technological vulnerabilities will be discussed in more detail in chapter 4.3 Due to these risks, a key element of data confidentiality is encryption. Companies might store their data unencrypted within their own security perimeter, but in the cloud that poses massive risks to data security. This is why strong encryption is vital, especially in the public cloud (Subashini, Kavitha 2011, 4).

According to Grobauer, Walloschek & Stöcker (2011, 52), web services and cryptography, which are essential technologies to the cloud, have multiple vulnerabilities. Web technology is vulnerable in regard to session handling; and the array of different web protocols pose many vulnerabilities, which could lead to data leaks. The risks presented by the vulnerabilities in web technologies will be more thoroughly discussed in chapter 4.3.2. Data confidentiality could also be compromised due to nonexistent encryption, or because of the breach of a weak cryptographic algorithm. In addition to the access automated systems have to the data, the cloud service provider might also have elevated privileges to the cloud in order to do their job. These privileged accounts present yet another threat to the confidentiality of the company data. These vulnerabilities make the auditing and monitoring of data creation, modification and deletion a vital part of the cloud security (Takabi, Joshi & Ahn 2010, 26). A closely related theme to data confidentiality is data privacy; the protection of the former also protects the latter and vice versa. Data privacy will be discussed separately in chapter 4.2.5.

### *4.2.2 Data integrity*

The second component of data security is data integrity. It can be achieved by ensuring that all processes that modify the data take into consideration the atomicity, consistency, isolation and durability of the data, also referred to as ACID (Vogels 2009, 41–42). To ensure this, it is imperative that only those who are authorized can access the data and modify it. It is equally important to differentiate in which ways particular users can modify the data; some users have more privileges than others (Zissis, Lekkas 2012, 586). A prerequisite for preserving the integrity of the data is rigorous access control along with proper handling of data backups (Kaufman 2009, 62).

According to Subashini et al. (2011, 5), a stand-alone database can ensure the ACID and thus the integrity of data with the help of different constraints and transactions in the database. The process becomes much more complex in the cloud. Distributed systems use transaction managers to ensure the data integrity during transactions and modification of data, but this doesn't suffice in the cloud environment. Vogels (2009, 40–41) argues that the complexity and the high availability of the cloud combined with the massive amounts of requests present many technical challenges to the consistency of the data; sometimes trade-offs must be made between durability, availability and consistency of the data.

Browser access is the modus operandi of the cloud. It could create problems with the management of the data integrity, as some web protocols such as HTTP or HTTPS (Hypertext transfer protocol) don't support transactions. This might lead to the integrity management having to be dealt with on the API level (Subashini, Kavitha 2011, 5). Due to the lack of standards, this approach is problematic. It might lead to situations where the data integrity is ensured on the database level but the application level is entirely surpassed in the process. Depending on the internal logic of the application, this might present serious problems regarding data integrity (Subashini, Kavitha 2011, 5).

### *4.2.3 Data availability*

The third component of data security is the availability of data. Being able to access business critical data at all times is one of the biggest concerns companies have when considering moving their data and applications to the cloud (Fox et al. 2009). In short, availability is the system's ability to provide data and software to the users reliably regardless of actual demand. By this definition, the high fluctuations in demand have to be accommodated by the cloud (Zissis, Lekkas 2012, 586 – 587). Venters & Whitley (2012, 186) argue that it is justified to expect the availability of the cloud to match or even exceed the availability of the equivalent in-house infrastructure. The high availability is made possible by a thorough analysis of the actual demand for the cloud services.

Jansen & Grance (2011, 31) define the availability as the accessibility and usability of the organization's resources. It could be temporarily reduced by way of a DoS attack, broken hardware or natural disasters; the aspect that causes the most worry to the organization is the unpredictable nature of these threat events.

Usually the most concrete manifestation of the cloud computing availability is the service level agreement, or SLA, where the required weekly or monthly uptime of the service is defined (Ramgovind, Eloff & Smith 2010, 4). The availability needs to be taken into account already in the development phase, as the applications need to support high availability and scalability. The infrastructure needs to be multi-tiered, which is made possible by instances of virtual machines dispersed on different servers, managed by a load balancer. This significantly increases the resiliency of the infrastructure, and if such measures are not in place, it might severely undermine the availability of the system (Subashini, Kavitha 2011, 7). The dispersion of resources on different servers within one data center helps to protect critical systems from hardware failures, but the datacenter itself is still vulnerable to external threats. Stoneburner, Goguen & Feringa (2002, 13–14) categorize common threat sources into natural threats, human threats and environmental threats. Natural threats include floods, earthquakes and the like. Human threats could be either intentional such as attacks over the network or planting of malicious software; or unintentional like accidental removal of data. Environmental threats include power failures, pollution or leakage into the data center. Large providers like Amazon and Google offer their services based on multiple countries and continents, which significantly mitigates the effects of bigger outages that are a result of natural disasters or other unexpected events (Zhou, Zhang, Xie, Qian & Zhou 2010, 107).

Jensen, Schwenk, Gruschka & Iacono (2009, 114) discuss different cases of flooding attacks that could lead to the cloud service becoming unavailable. Direct denial of service attacks (DDoS) could utilize a vulnerability specific to cloud computing; namely, some services having only a single entry point. Once this entry point becomes flooded because of a direct attack and is rendered unavailable, the whole service could be down although there would still be available resources elsewhere in the cloud (Jensen et al. 2009, 115). Indirect DoS present another kind of vulnerability that has its basis in the very nature of the cloud: utilizing the same infrastructure for multiple companies. An attack that targets one company could inadvertently affect the services of another company on the same cloud. The sophistication of the cloud technology could also pose an additional threat: some cloud services might try to automatically move the workload onto another server after the load balancer notes the increased load on the attacked server. If it succeeds, it only escalates the problem further, even though that was not the intent (Jensen et al. 2009, 115).

### *4.2.4 Data location*

One aspect of data security is the data location. Despite that, the cloud customer is often unaware of the exact location of their data, even the country in which the data resides might be unknown (Popović 2010, 347). The data location is especially relevant when sensitive and personally identifiable information (PII) is at stake, as such data is subject to stringent regulation regarding storage and deletion in many countries (Subashini, Kavitha 2011, 5). A good example of such regulations is the European Union (EU) regulations, which state that PII data need to reside within the geographical borders and thus within the jurisdiction of the EU. The regulations also place further restrictions are also on data transfer out of the EU area (Brender, Markov 2013, 728).

In addition to regulation regarding data citizenship and residence, national laws can pose bigger threats to an organization's data. For example, data seizure might be a realistic scenario if the data resides in a country where the legal framework allows such measures to be taken (Popović 2010, 345). Brender & Markov (2013, 728) also discuss the risk of such legislation that gives the government a vast access to an organization's data, they mention the United States (US) Patriot Act as an example of such law. It gives the Federal Bureau of Investigations (FBI) broad access to an organization's data, provided that a court order is presented. This might have far-reaching consequences to the CIA of an organization's data.

Besides having a big legal impact, the data location also presents some practical challenges and implications to the cloud customer, specifically regarding the due diligence in the cloud. On-site inspection of the physical location of the data cannot be carried out if the data center is located in another country; performing an audit on a third-party provider might also prove to be tricky (Kalyvas, Overly & Karlyn 2013, 20).

As discussed in chapter 4.2.3, many cloud service providers disperse the data to multiple locations to achieve resiliency against different kinds of incidents and failures. According to Zhou et al. (2010, 111) this further exacerbates concerns in the cloud, as the applicable regulations are not clear to all parties. This regulatory complexity is especially true if there are multiple parties involved in providing the cloud service, e.g. a SaaS-vendor running their applications on top of Amazon's cloud, instead of just one service provider being responsible for the whole service. The regulations don't necessarily take into account this complexity, which might risk the confidentiality of an organization's data in case they end up in a courtroom (Zhou et al. 2010, 110). Thus, it is not irrelevant where the data is geographically located, even though it would have little impact on the functionalities and services in the cloud.

### *4.2.5 Data privacy*

Privacy is an elemental part of cloud computing security, and it's an underlying theme in many discussions regarding risks in cloud computing. Katzan (2010, 8) defines privacy as the right of an individual or an organization to decide how, when and how much information about them is available to other parties. Mather et al. (2009, 146) further define this information, or PII, as any information that is related to a known or identifiable individual. To better understand the dynamics of the privacy in the cloud, Katzan (2010, 8) discusses different actors that have a role in the privacy domain in the cloud:

- Subject
- Beneficial user
- Agency

Subject is the natural or legal person whose data is being handled or stored; the beneficial user is the party that gains value from the data, and the agency is the system or systems that process the data. The agency is neutral in its stance to the data; it merely performs the processing, whereas the subject and beneficial user both have an interest to the data. The subject and the beneficial user might, but don't necessarily belong to the same organization (Katzan, 2010, 8).

The subject, whether a natural or a legal person, usually wants to carefully control the privacy of their data, whereas the beneficial user might have an opposite incentive in order to maximize their possibilities for utilizing the data. This creates a conflict of interest, which is the biggest in the free-of-charge consumer-grade SaaS offerings; cloud customers get free services by allowing the provider to utilize their data. However, as more organizations adopt such services to replace their in-house email servers or other business applications, this issue is very relevant for this study as well.

According to Chakraborty, Ramireddy, Raghu & Rao (2010, 34), the risks related to privacy are a major concern in the cloud alongside security and business integrity. The cloud customers generally worry about the adequacy of privacy controls in the cloud, as they have only limited means to monitor and assure the privacy practices of the cloud vendors. The traditional approach to protecting data in information systems is to simply manage the access to the data, the methods for which will be discussed in chapter 4.3.3. These measures are still very relevant in the cloud, but in order to protect data privacy they alone are not sufficient.

Due to the sensitive nature of PII, the cloud privacy is subject to many rules and regulations over the world. Zhou et al. (2010, 111) provide examples of the US regulation, such as the Electronic Communications Privacy Act (ECPA), which regulates governmental access to electronically stored communications, and the Gramm Leach Bliley Act (GLBA), which regulates the disclosure of financial information. These and many more US regulations explicitly limit certain use purposes of the data but otherwise give broad freedoms to cloud service providers. The EU regulations, on the contrary are based on certain undisputed principles, which strictly limit the use of PII and its transportation outside the borders of the EU (Mather et al. 2009, 155). A good example of such regulation is the EU-ruling also known as the “right to be forgotten”. It’s a 2014 ruling by the EU Commission regarding the privacy of personal data, which mandates the cloud service providers to remove links to publicly available PII by request of the individual to whom it concerns. The requests can be placed once an individual considers the PII to be “inaccurate, inadequate, irrelevant or excessive” (Factsheet on ruling C-131/12, 2–3).

The low degree of control in the cloud places additional requirements on the data privacy while the risk of unauthorized access is increased; the appropriate privacy protections need to be incorporated in all cloud security measures (Takabi, Joshi & Ahn 2010, 28). Due to the obfuscation of infrastructure boundaries and the number of different actors in the cloud, there is a need for more sophisticated methods of protecting the privacy of the data. Instead of protecting data only from outside access, the data should also be protected from within (Chow, Golle, Jakobsson, Masuoka & Molina 2009, 88).

A data breach resulting in the loss of privacy can lead to significant adverse financial and reputational consequences to an organization (Mather, Kumaraswamy & Latif 2009, 145). The protection of the data from within, as proposed by Chow et al. (2009, 88), can be achieved by using encryption. Encryption is a very effective tool for protecting the privacy of the data, but this effectiveness negatively affects the usability of the data, as performing computing operations, especially searching and indexing the data, is difficult when the data is encrypted.

Yu & Wen (2010, 1) argue that the data security should be understood beyond the triad of confidentiality, integrity and availability or data storage and transmission security; instead the whole data life cycle should be taken into consideration. They propose the following concept for the life cycle of the data:

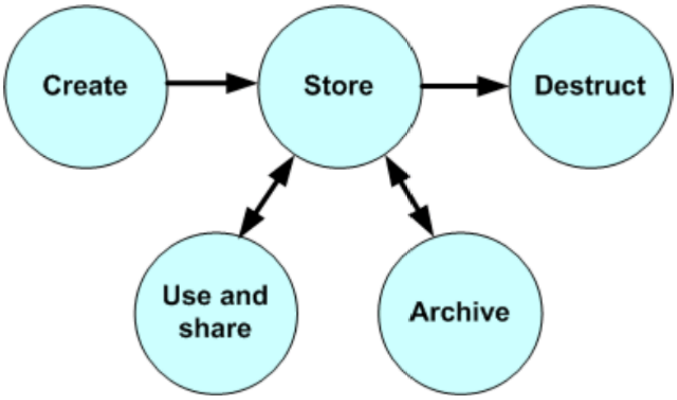

Figure 4 Data life cycle (Yu, Wen, 2010, 1)

In this model, the life cycle of data begins with the creation and thereafter the storage of the data. Once in storage, the data could be accessed and shared multiple times. The archival of data is used to store data that is not in active use but might still needed in the future. The final step of data life cycle is the destruction of data (Yu, Wen 2010, 2). This life cycle thinking could ease the data security management and data privacy, as by implementing data security functions an organization could get better assurance to the CIA of the data, especially after the data has been removed (Yu, Wen 2010, 3–4).

Indeed, the end of the data life cycle, i.e. the destruction of the data could prove to be problematic, especially in regard to the data privacy. Mather et al. (2009, 149) discuss data retention and data destruction as big concerns regarding data privacy. The retention refers to the ownership of the data during its lifecycle; who owns the data and what happens to the data once the agreement between cloud customer and cloud service provider ceases. The data destruction is closely linked to data retention, as it is the next logical step in the data life cycle. However, it’s not always so easy to ensure that the data is truly destructed. The deletion of the data from a hard disk only releases the block where the data was stored instead of really removing the data. Overwriting the block, several times, could render the old data unusable; but there’s no guarantee of that (Mather et al. 2009, 153). Data remanence is a term that refers to what is left of the data after it has nominally been erased; it is the residue of the data. It could pose a serious threat to the data privacy, and as the cloud customer has no control over the physical infrastructure, the assurance of the actual destruction of the data is difficult (Zissis, Lekkas 2012, 586). Due to the multitenant nature of the cloud, the physical resources are shared between the cloud customers. Once the resources are no longer in use, they are released back into the resource pool to be used again, a practice called object reuse. This might enable malicious actors or other cloud residents to gain access to the residual data and thus compromise the privacy of the data (Zissis, Lekkas 2012, 586).

The most certain way of destructing the data is the scrapping of the hardware that is no longer used. This is the cloud service provider’s responsibility, and although it is an industry standard to thoroughly erase data from hardware before it’s decommissioned; the customer has no way of ensuring that the hardware and thus the data have truly been destructed. This is why media sanitization by way of destructing hardware is not sufficient or practical in the cloud context. To prevent the data ending up to the wrong hands, additional cryptographic measures are required (Grobauer, Walloschek & Stöcker 2011, 53). This way the access to the confidential data doesn’t compromise the privacy of the data, as the encrypted data is unintelligible to all parties without the right cryptographic keys (Mather et al. 2009, 153).

## 4.3 Technological security risks

The different cloud technologies have been an underlying theme throughout the whole study. To further clarify: there is no isolated or clearly definable concept of cloud technology; the cloud consists of a combination of different, older and newer technologies that as a whole form the cloud. As the focus in study is on the risks in cloud computing in general (RQ1) and their relevance to different service models (RQ2), also the technological risks play an important role. However, as this is not a study in the area of technology but in the area of business, a very detailed insight into the technologies is out of scope. Thus the following subchapters present the risks related to the most crucial technologies enabling cloud computing, namely virtualization, web technologies and identity and access management technologies.

### *4.3.1 Virtualization risks*

As shortly introduced in chapter 3.1, Virtualization is the technological backbone of cloud computing, making the dynamic provisioning of resources possible (Tsai et al. 2012, 32). Virtualization is made possible by hypervisor, also known as a virtual machine monitor (VMM), which provides the technological platform for hosting virtual machines. Simultaneously, the hypervisor also enables multitenancy, which is defined as the running of multiple virtual machines on a single host computer (Brown, Anderson & Tan 2012, 1). Hypervisors can be divided into two groups based on their build type. Type 1 hypervisors are so-called bare-metal hypervisors, which access the hardware resources directly. Type 2 is a virtual hypervisor, which runs on atop an operating system (Mather et al. 2009, 45–46).

Due to its location at the bottom of the cloud stack, closest to the infrastructure, the hypervisor faces very stringent requirements regarding security. All processing operations performed in the cloud will be routed through the hypervisor, which make it a very significant single source of risk (Robinson et al. 2010, 29). The cloud customer has no visibility or access to this level of the cloud infrastructure, and they rely on the cloud service provider for the operations and maintenance.

According to Vaquero, Rodero-Merino & Morán (2011, 7). As the hypervisor includes significantly less code than a full-fledged operating system, in theory its security is easier to verify and maintain than that of an operating system. However, the rapid development of new cloud offerings has made the VMMs somewhat more complicated, thus also complicating the security management.

Vaquero et al. (2011, 3) analyze different security concerns to cloud computing with a specific focus on the virtual machines and networks. The key security concerns they found were the security of the virtual machine (VM) images and the hypervisor. Indeed, the security of a single virtual machine in the cloud is crucial, as there are multiple different vulnerabilities that could stem from a compromised VM.

Tsai et al. (2012, 34 – 35) present four different security issues present in virtualization technology:

- VM hopping
- VM mobility
- VM diversity
- VM denial of service

VM hopping means the use of one VM for gaining an illicit access onto another VM. Ristenpart, Tromer, Shacham, & Savage (2009, 2) studied two different threat scenarios that utilize vulnerabilities in the virtualization layer for VM hopping. The first scenario is a malicious actor attempting to compromise the data of any cloud customer without a specific the target; the second scenario specifies the target. They concluded that by using sophisticated probing, an attacker might be able to locate a particular cloud customer with high certainty and arrange so that their virtual machine is located on the same physical device. This in turn enables gaining unauthorized access to cryptographic keys or performing an internal DoS attack. Ristenpart et al. (2009, 2 – 3) tested an access method called a side-channel attack, which utilizes information that is available without significant effort. It is also noteworthy that they only utilized the capabilities that were readily available in the cloud.

VM mobility is the act of transporting a virtual machine from one physical host onto another host (Tsai et al. 2012, 35). This transportation process could be subject to a man-in-the-middle -attack, potentially compromising the confidentiality and integrity of the data. If the data is not properly encrypted, there are risks of the data being leaked or modified during the transport (Vaquero et al. 2011, 6). Due to the large scale of the cloud, the spreading of compromised VM images could greatly exacerbate the effects of this vulnerability (Tsai et al. 2012, 35).

VM diversity refers to the various different virtual machine configurations in the cloud. Many traditional security mechanisms for virtualization work with the underlying assumption that the virtual machine image, also known as guest operating system is known and trusted. However, as only the cloud customer has control over the image, this is often not the case (Tsai et al. 2012, 32). The process of managing and transporting virtual machine images to the cloud over insecure networks presents threats to the integrity of the OS image (Vaquero et al. 2011, 16).

The last threat presented by Tsai et al. (2012, 35) is the VM DoS. In its simplicity, one VM uses up all the available resources, thus rendering all the other VMs residing on the same host unusable.

These vulnerabilities make it clear that the isolation and security management of virtual machines is an essential part of secure cloud computing. On a shared infrastructure there is always the possibility that a party with malicious intentions is sharing the same physical resources. If a virtualization-related vulnerability is discovered, the malicious party could gain unauthorized access to the cloud and so threaten the confidentiality and integrity of the company data (Ristenpart et al. 2009, 2).

Robinson et al. (2010, 29) identify two main areas of concern in virtualization security: problems related to the isolation of vulnerabilities in configuring and deploying virtual machines; and the applicability of the current security controls for virtualization in the cloud environment. Brown, Anderson & Tan (2012, 8) recognize various security threats stemming from virtualization: data isolation, side-channel attacks, unknown and potentially malicious actors sharing the same infrastructure, uncoordinated changes made in the infrastructure and the potential mixing up of data of different cloud customers.

VM escape is an event where a virtual machine misconfiguration allows access to the physical host, which in turn enables running arbitrary code on the lowest level of the cloud stack. This way an attacker could bypass all the security measures in place on all levels of the cloud stack, creating a uniquely powerful security threat to all those using the same infrastructure (Krutz, Vines 2010).

It could be concluded that virtualization presents many technical threats to cloud computing. It also presents many such threats that are not attributable to the relationship between the cloud service provider and the cloud customer. Instead, the threat comes from the direction of the other cloud customers. As the cloud customer has no way of knowing who is sharing the infrastructure with them, the security measures need to be in place to maintain confidentiality, integrity and availability of company data (Ristenpart et al. 2009, 1).

### *4.3.2 Risks in web technologies*

Ubiquitous access is present in all cloud service models, as defined in the essential characteristics of cloud computing. In practice this means that the services are always accessed over a network connection using a web browser (Subashini, Kavitha 2011, 6). As discussed in chapter 4.1, certain vulnerabilities can be considered cloud-specific when they are a part of key cloud computing technologies or have their cause in one of the essential characteristics of the cloud, and web technology fulfills both of these criteria (Grobauer et al. 2011, 52).

Web technologies are an essential part of the cloud, and they include many vulnerabilities that present risks to cloud computing. Jensen et al. (2009, 111 – 115) discuss these security concerns, among other technological security concerns, listing XML signature vulnerabilities and browser security as major concerns. Subashini & Kavitha (2011, 6) present a breakdown of security breaches based on the layer where the breach occurs; application layer was compromised in 39% of the cases.

The Open Web Applications Security Project (OWASP) lists the top 10 security concerns related to web technologies, periodically updating the list according to the development of the risk landscape. The aim of the project is to raise awareness of the most common security risks, thus improving the web security as a whole (OWASP 2013, 1–2). This list is a comprehensive overview of the risks in web technologies, the theme of this chapter, and these risks are well applicable to cloud computing (OWASP 2013, 4):

- Injection vulnerabilities (SQL, LDAP)
- Broken authentication and session management
- Cross-site scripting
- Insecure direct object references
- Security configuration problems
- Exposure of sensitive data
- Missing access control on functional level
- Forged cross-site requests
- Usage of vulnerable components
- Invalid redirects and forwards

The injection vulnerabilities are caused by untrusted data being incorporated into a legitimate command that is executed, potentially causing multiple problems. Authentication and session management risks might lead to the compromising of passwords or keys that could be used in combination with other vulnerabilities, whereas cross-site scripting tricks the browser to use untrusted data, possibly leading to session highjacking or redirect to a malicious web page. Insecure object references might expose files or database keys to external parties; security misconfigurations often result from neglecting the defining and maintaining of security configurations and software patches and updates. Sensitive data exposure is possible if the data is not properly protected; as discussed in chapter 4.2, different security measures are needed to ensure the CIA of the data. Access control on a functional level is required to ensure that the commands executed on the server side are valid; otherwise requests could be forged on the browser, further threatening the data security. Cross-site request forgery allows a malicious actor to generate requests from a victim's browser, again compromising data security. Using vulnerable libraries, frameworks or software modules poses a serious threat, as it might enable an attacker to take over a whole server. Finally, the last on the list, invalid redirects and forwards could direct the user to a malicious web site, creating a possibility of phishing or malware infection (OWASP 2013, 5–15).

This is by no means a comprehensive list of all web service vulnerabilities, there are many more risks and vulnerabilities present in web technologies. The list gives a good overview of the complexity and nature of the vulnerabilities that are an inherent part of the web technologies used in cloud computing. Many of the listed risks are related to web browser technology and web service frameworks, and Jensen et al. (2009, 114–115) suggest that the strengthening of browser and framework security would greatly improve web security in general. Mather et al. (2009, 49–50) discuss the inadequacy of traditional perimeter security combined with network-based access controls, concluding that the approach works in a controlled environment but not in the public cloud where the risk level is significantly higher. To ensure web application security, the security measures need to be incorporated into the application design from the start.

### *4.3.3 Risks in identity and access management*

Identity and access management or IAM is a concept that enables the identification of users and managing their access to resources. In addition to providing access, the other and equally important part of the IAM is to prevent unauthorized access (Jansen, Grance 2011, 25). IAM also provides the capabilities for authentication, authorization and auditing. Authentication is the process of verifying the identity of the user, whereas authorization verifies the privileges of the user. Auditing is the process of monitoring the authentication and authorization processes; it has a critical role in detecting unauthorized access and possible breaches (Mathers, Kumaraswamy & Latif 2009 76–77).

Subashini & Kavitha (2011, 8) divide the identity management into three subcategories:

- Pure identity paradigm
- User access paradigm
- Service paradigm

The pure identity paradigm entails only the creation, management and disposal of identities. No stance is taken on the access rights these identities have. The user access paradigm entails a user and a simple access to a system, and the service paradigm grants role-based online access to multiple services (Subashini, Kavitha 2011, 8). The first two paradigms might be too coarse for the cloud as they make the principle of the least privilege difficult to implement. Without fine-grained assignment of rights, the risk of exposing confidential data is increased (Grobauer, Walloschek & Stöcker 2011, 57). A role-based separation of access rights is widely regarded as the best practice for capturing various requirements and policies under one access paradigm. It also ensures that the principle of the least privilege can be implemented (Takabi, Joshi & Ahn 2010, 27–29).

Traditionally the organization has a clear boundary of trust, which is controlled and secured by the IT organization. The adoption of cloud computing challenges this clarity, as it extends the boundary over to the cloud service provider (Mather, Kumaraswamy & Latif 2009, 73). Identity and access management is a requirement for safe and secure cloud computing, so the IAM process needs to be extended to the cloud too. The easiest way to implement IAM to the cloud is to simply add another, independent IAM process for the cloud (Jansen, Grance 2011, 25). This duplicate maintenance of user identities and access rights is a risk source to an already complicated process (Mather, Kumaraswamy & Latif 2009, 76).

Another way of implementing the IAM process to the cloud is the synchronization of credentials between the enterprise infrastructure and the cloud (Subashini, Kavitha 2011). However, leveraging the existing IAM framework in the cloud is often difficult, as the cloud resides outside the company perimeter and because there is a lack of technical compatibility (Jansen, Grance 2011, 25). Different organizations that operate on the cloud have different security and privacy requirements, which necessitates the implementation of various security mechanisms to fulfill these requirements (Takabi, Joshi & Ahn 2010, 26–27). Without proper orchestration, architecture and central governance the IAM process is prone to mistakes and poor quality of user data. High personnel turnaround, external partners and big amount of different user identities and procedures make the up-to-date management of identities a challenging process (Mather, Kumaraswamy & Latif 2009, 74).

A third way of taking care of the IAM process is federated identity management (Subashini, Kavitha 2011, 8). This approach requires reliable and careful handling and separation of the identities to protect the data and to prevent the mix-up of cloud provider and cloud customer identities. The federated IAM could be implemented within the company perimeter or acquired from a service provider. The first approach lets the customer utilize the existing IT infrastructure and security measures, but it might cause overhead for maintaining identities that are not employees. The outsourcing approach is flexible especially if there are interfaces between corporate IT architecture and multiple different service providers, but the downside is the lack of transparency and control (Mather, Kumaraswamy & Latif 2009, 94–98).

According to Grobauer et al. (2011, 56–57) the biggest risk sources in IAM are weaknesses in credential reset mechanisms, ineffective authorization checks, too coarse authorization and insufficient monitoring possibilities. Credential resets provide an avenue for attackers to gain access to the cloud. Ineffective authorization checks pose risks to the confidentiality of data, as they might make data available to unauthorized parties using methods such as URL guessing. Coarse authorization control makes the separation of duties impossible. This leads to a risk of unauthorized users gaining too high privileges and thus gaining access to the cloud management interface. Finally, the lack of proper monitoring and reporting makes the identification of all other threats difficult, which is very problematic to the information security of the cloud.

## 4.4 Relationship risks

In addition to the various technological and data security risks that are usually discussed when cloud computing is considered, there are organizational risks in the cloud as well. In essence, cloud computing is a form of outsourcing; the risks related to outsourcing apply equally to cloud computing. This chapter elaborates on these organizational aspects, and later in chapter 6 various approaches for managing the cloud relationships are presented.

### *4.4.1 Lack of trust*

Trust is a key element in any business relationship, but even more so when an organization decides to transfer their business-critical data outside their safety perimeter and into the hands of a third party. Khan & Malluhi (2010, 20) define trust as the level of confidence in someone acting or behaving according to expectations; it's a metric of reliability. Garrison et al. (2012, 66) analyzed the relationship between the cloud service provider and cloud customer, and concluded that in order to successfully adopt cloud computing, mutual trust must first be established. They define trust as the expectation that the other party performs their responsibilities well and treats the other organization responsibly. Although they analyzed the relationship mainly from the customer's perspective, this definition works both ways and can be applied to both parties. Without trust, the interaction between the parties could be defined by uncertainty and suspicion, possibly leading to miscommunications (Garrison, Kim & Wakefield 2012, 68).

Sangroya et al. (2010, 260–261) discuss the concept of trust and its relationship with the cloud technologies, noting that the current security concepts such as secure socket layer (SSL) combined with signatures and authentication methods do not alone guarantee security in the cloud. Andert, Wakefield & Weise (2002, 8–10) divide the concept of trust into three different dimensions:

- Direct trust
- Transitive trust
- Assumptive trust

Direct trust is applicable when the authentication and security assurance process are performed by one single party; this is the most stringent model of trust and it's usually employed by organizations that deal with sensitive data, such as health care organizations, insurance companies or financial entities. This trust approach requires the most effort from the organization, as it is solely responsible for performing the validation of credentials. Transitive trust enables the partial delegation of the authentication and security assurance process to multiple parties, given that all these parties have aligned their security policies and processes. This model of trust simplifies the validation process, as there are multiple responsible parties and the burden can be shared. Assumptive trust provides the least amount of trust of these trust models; there is no credential validation. As the name hints, the model works wholly under the assumption of trust, no more; this is why its application into the business domain should be carefully considered (Andert et al. 2002, 10).

Ko, Jagadpramana, Mowbray, Pearson, Kirchberg, Liang & Lee (2011, 2) discuss different controls of trust, dividing them into preventive and detective controls. The former are in place to prevent risks from realizing, and include the usual security measures in the cloud; the detective controls complement the preventive in the sense that they provide information in case the preventive controls have failed. Both controls are required to successfully manage the trust in cloud computing.

Khan & Malluhi (2010, 23) discuss various challenges to trust, naming diminished control and lack of transparency as major challenges. As discussed earlier in chapters 3.3 and 3.4, the use of cloud computing resources inevitably leads to less control over computing resources, and this creates challenges regarding trust. The extent to which an organization loses control of the resources and delegates it to a service provider depends strongly on the cloud service and cloud deployment models; the transparency is similarly dependent on the chosen models.

Finally, Kumar, Sehgal, Chauhan, Gupta and Diwakar (2011, 419) conclude that the level and also the importance of trust in cloud computing depends on the participating organizations; it also depends on the value of the data that is stored and processed in the cloud. The less trust there is between the parties, the more control the cloud customer desires; all the way to the level of technology which would otherwise be delegated to the service provider. To conclude, there is always a balance between risk and trust; the more trust, the less risk and vice versa.

### *4.4.2 Lock-in*

Cloud computing forms a relationship between the cloud service provider and the cloud customer. This relationship is and will be a great source of risk. The magnitude and the type of the risks posed by the relationship depend on the level of trust between the parties in the cloud and the level of commitment in the relationship. Trust could be defined as the confidence between the parties, based on how well the values of both parties are aligned (Garrison, Kim & Wakefield 2012).

This relationship is not always balanced, as sometimes one party could have leverage over the other. The provider has great influence over the customer once their data is on the cloud, as it might be difficult and costly to get it transferred onto another infrastructure. This balance is prone to change over time, for example if new cloud providers come to the market and increase competition.

One big potential risk in the relationship between the provider and customer is a situation called lock-in, often referred to as vendor lock-in or data lock-in. It refers to the situation where the vendor has power over the cloud customer, as the customer cannot easily change the service provider and migrate their data and applications to another service provider's cloud (Fox et al. 2009, 15). The lock-in has its grounds in the proprietary technology, which leads to lack of interoperability and thus makes it very difficult to transfer the corporate data over to another service provider. The lack of interoperability is in practice the lack of standardization, which greatly hinders the development of the cloud computing market (Penzelm, Kryvinska, Strauss & Gregu 2015, 393).

The vendor lock-in has different consequences in different cloud service models. In SaaS, the lock-in concerns mainly the data. It is usually stored in a proprietary database, which prohibits the simple export and import of the data. Highly customized cloud applications might lead to the need to re-structure the data to make it compatible with another service provider's applications. This leads to very high costs when switching the provider (ENISA 2009, 26). This is also where the descriptive standards would help the mobility of the data (Borenstein, Blake 2011, 75).

In PaaS, the lock-in takes place on the level of the API. This hinders the portability of the applications, as the runtime environments in PaaS are often highly customized and require extensive modification to existing applications to make them functional on another platform than what they were built on (ENISA 2009, 27).

Finally, the lock-in in IaaS is very dependent on the type of services provided, and it is largely a question of virtual machine portability and compatibility. The amount of data is also a factor; the risk of lock-in increases linearly with increase in data quantity (ENISA 2009, 27).

The vendor lock-in dilemma could grow even bigger with the ever-expanding cloud offerings; a saturated market combined with a big number of providers makes the differentiation between different providers hard (Brender, Markov 2013, 729). The vested interests of the cloud service provider might also contribute to the problem of vendor lock-in, as the providers might have more incentives to hold on to their customers than make their data easily portable (ENISA 2009, 25). The risk of bankruptcy of a service provider could pose a serious threat to business continuity and jeopardize the company data (Sultan 2011, 275).

### *4.4.3 Outsourcing risks*

Clemons & Chen (2011, 3–6) compare the risks in cloud computing to risks in traditional outsourcing. In an IDC survey, the main concerns in outsourcing relationships were lock-in resulting from reduced competition and leading to cost increases; reduced control over the resources, specifically intellectual assets and finally, performance, data security, integrity and availability. Opportunity risk is such behavior where the provider acts out of self-interest and operates in a way that harms the customer. Clemons & Chen (2011, 4) list three main sources of opportunity risk:

- Shirking and subpar performance
- Poaching and intellectual property theft
- Opportunity pricing, client lock-in and vendor hold-up

Shirking refers to the situation where the vendor charges for full services but in reality delivers subpar performance; this is only possible because of asymmetric information. The motivations could be financial, such as postponing or completely avoiding investments in infrastructure. Poaching could be considered a more serious threat to an organization, as it involves the compromising of the data and the potential resale of confidential data to a competitor. Opportunistic re-pricing is in principle the same as vendor lock-in, as it stems from the vendor gaining leverage over the customer in the negotiations and thus being able to charge high prices for the services as the costs of changing provider would be even higher (Clemons & Chen 2011, 4). Such risks are also discussed by Kaliski Jr & Pauley (2010, 4), who note that the tough competition in the cloud environment and the reduced visibility into the security and privacy arrangements of the cloud make it a tempting option for cloud service providers to make compromises to achieve cost savings. As a solution they call for more visibility into the monitoring and assessment of cloud computing, these topics will be elaborated on in chapter 6.3.

Chow et al. (2009, 86) discuss the issue of third-party data control, raising multiple concerns over the applicable regulations and the lack of transparency and control. They raise the following points:

- Due diligence
- Auditing
- Contractual obligations
- Espionage
- Data lock-in
- Transitive relationships

Due diligence questions are raised when a legal authority requires data from an organization; they need to be able to respond in a timely manner but storing the data in the cloud might hinder this process. Auditing is made difficult by the lack of control, as already discussed in the previous chapter; despite that the process of auditing must be carried out. The contractual obligations organizations face when signing cloud agreements need to be assessed properly, as sometimes these agreements include overzealous and overly broad statements. A good example is the Amazon's EC2-contract, which technically prohibits the customer from filing an intellectual property infringement claim against Amazon, ever.  Espionage is a problem, especially in industries that have a high degree of intellectual property. There is always a risk that the cloud vendor unlawfully utilizes the data that is stored on their infrastructure for malicious purposes. Data lock-in or vendor lock in was discussed in the previous chapter. Last, the transitive nature of cloud computing relationships might lead to situations where the cloud service provider uses a subcontractor for running the infrastructure; thus further lowering the level of control in the cloud (Chow et al. 2009, 87).

The organizational and relationship risks discussed in this chapter are only one subset of the risks, but as the examples presented in the chapter show, they are a fundamental contributor to the overall risk in cloud computing. For technological risks there are technological solutions, but to properly implement these solutions and to monitor their success, the relationships need to be properly taken care of. Some of the risks in cloud computing can be managed more easily than others; some are more complex and require more nuanced risk management approaches. An example of a more straightforward risk category are the technological risks; they are caused by technology and can usually also be mitigated or minimized with the application of proper technological solutions. The organizational or relationship risks, on the other hand, require more refined and multifaceted risk management approaches and solutions.

# 5 RISKS IN CLOUD SERVICE MODELS

The discussion about the different risks in previous chapter didn't take any stance to the relationship between the different risk categories and the cloud service models. However, as specified in the RQ2, this relationship is of great interest to this study, and the aim of this chapter is to analyze this relationship and the relevance of different risks and risk sources to each of the cloud service models.

## 5.1 Risks in IaaS

As discussed in chapter 3.3.1, IaaS is the service model that has the lowest level of abstraction to the cloud customer. This has an implication to the risks that can be considered most relevant to IaaS; most of them are on the lower end of the cloud stack. The responsibilities for the security measures in the cloud are shared between the cloud service provider and the cloud customer, the service provider being responsible for the security of the hypervisor and the infrastructure below it, the customer having the responsibility from the OS level upwards. In practice this means that the one of the main sources of risks in IaaS is the virtualization technology (Subashini, Kavitha 2011, 9). As Mather et al. (2009, 45) discussed, the risks in virtualization could be further categorized into risks in virtualization software and risks in virtual OS. The risks presented in chapter 4.3.1 by Tsai et al. (2012), namely VM hopping, VM mobility, VM diversity and VM denial of service, are all very relevant in IaaS. Vaquero, Rodero-Merino & Morán (2011, 5–7) also discuss the relevance of different virtualization-related risks in the context of IaaS; they list VM monitoring, side channel attacks, data security and network virtualization as key risks in IaaS.

According to Mather et al. (2009, 135-136) the cloud customer's responsibilities in IaaS include the management of VM images including their security management, OS-level security, database security and finally, access control management. This means that the IAM process, discussed in chapter 4.3.3 is also the cloud customer's responsibility. The risks related to data security are also relevant to IaaS. The confidentiality, integity and availability, as well as the location and privacy of the data could be compromised.

The risks specific to IaaS are, as stated in the beginning, on a low level of the cloud stack. The applications that are hosted on the infrastructure can present a trove of risks; however, they are not related to the cloud per se. Because the view on the risks in this chapter is limited to only those specific to the service model under scrutiny, the risks posed by the application that is running on top of what's considered in scope of the IaaS are irrelevant. For example, the risks posed by web services are an equally serious risk regardless of the service model; however, these risks are not related to IaaS but rather the applications themselves.

## 5.2 Risks in PaaS

In PaaS, the most significant risks are related to the application deployment environment, which is a key part of the cloud platform. Subashini & Kavitha (2011, 8–9) discuss the level of control the PaaS customer has over the security features and the relevant responsibilities. They argue that the PaaS provides more capabilities for the customer to create custom security solutions in addition to the security features provided by the development platform, but also note that developing such solutions might present further complexity and undermine the cloud security. Risks in web technologies are also relevant in in PaaS, as the development platform is browser-based and thus subject to all vulnerabilities that concern the underlying technologies (Grobauer, Walloschek & Stöcker 2011, 52). The risk of a data lock-in is relevant to PaaS, specifically when the platform utilizes proprietary technology (Fox et al. 2009, 15).

## 5.3 Risks in SaaS

As discussed in chapter 3.3.3, SaaS has the highest level of abstraction to the customer, which has a significant effect on the risks that are specific or relevant to SaaS. The single biggest risk is lock-in, usually caused by proprietary technologies that give the cloud service provider leverage regarding the transporting of data out of their system, as the data won't be usable as such in other systems and might need to be restructured to be used elsewhere.

The web application security is a concern in all service models, but it has the most relevance for SaaS, as the model has the most complexity and usually is the most reliant on different web applications (Subashini, Kavitha 2011, 7). The responsibility for the web application security is mostly the service provider's responsibility, excluding basic security related routines such as user and access control. The standard SaaS user access controls might not be sufficiently fine-grained, which needs to be taken into consideration when planning for the security in the cloud (Mather, Kumaraswamy & Latif 2009, 53). Wu, Lan & Lee (2011, 556–557) propose the lack of trust in data security and network security as relevant risks to SaaS.

Popović (2010, 344–345) further lists privileged user access, data location, segregation and recovery; regulatory compliance and long-term viability as risk sources in SaaS. Long-term viability in this context is a reference to data lock-in; a bankruptcy or a merger with an unknown third company could create the risk of data lock-in or vendor lock-in. All in all, SaaS could be concluded to be the riskiest service model to the cloud customer in the sense that it poses the biggest risk of lock-in while providing the least amount of control and visibility into the infrastructure. However, these risks are present also in the other service models, only to a lesser extent.

To conclude this analysis and to answer the RQ2, most of the risks are present in all cloud service models. The risks that are present at the lowest level of the infrastructure are also applicable on the higher levels. The risks in virtualization are present in SaaS as much as in PaaS and IaaS, posing a risk of compromising the organization’s data security. The higher on the cloud stack the cloud service is positioned, the less control is provided to the cloud customer and the less independence they have regarding the cloud security measures. Indeed, when choosing the service model, an organization should carefully consider what they want to achieve by utilizing cloud, and especially what safety concerns need to be specifically accounted for in the cloud. In IaaS, the cloud customer has a high degree of control over the virtualized resources, but they also have a similarly high degree of responsibility for all the security measures in the cloud, from the OS level upwards. This might be a concern to the some organizations, as one of the promises of the cloud is the ease of adoption and ease of use. The manifestation of risks differs depending on the chosen cloud service model. This means that the cloud service models do affect the risks in the cloud, but only to a certain extent. Any single service model doesn’t introduce any novel risks compared to the other service models, i.e. IaaS inherently includes the same risks as SaaS, but some of the risks are more relevant to IaaS than to SaaS, as discussed.

# 6 MANAGING THE RISKS IN CLOUD COMPUTING

The focus in the previous chapter was on the risks in cloud computing, and the RQ2. This chapter proposes ways to manage these risks, and aims to create a link between the various risks in cloud computing and the tools for mitigating these risks. The aim is to provide an answer to RQ3. The management of risks here refers to either minimizing the outcomes of potential risks or removing the risk altogether, with the ultimate aim being the insurance of business continuity. Ramgovind, Eloff & Smith (2010, 5) suggest that to manage the risks in cloud computing, the organization should consider the current and future risks to cloud compliance, and estimate how the adoption of cloud computing affects those risks. This could be accomplished by adopting standards, frameworks or other tools that aid in increasing cloud transparency and governance.

## 6.1 Service level agreements

The service level agreement, or SLA is an elemental tool in managing the relationship between the cloud service provider and the cloud customer. It is a document that is used to clarify the contents of the service and the responsibilities of the different parties in outsourcing relationships; it also defines the quality of service (QoS) in the cloud (Patel, Ranabahu & Sheth 2009, 2). The contents of the SLA should be defined in a way that guarantee enough details and granularity to the cloud customer but at the same are simple to evaluate and enforce (Dillon, Wu & Chang 2010, 31). According to Kandukuri, Paturi & Rakshit (2009, 518), the service level agreement should include detailed information regarding the following points:

- The extent of the agreed services
- Definition of the agreed performance
- Agreement on problems resolution
- The responsibilities of the customer
- Guarantees and remedies
- Security measures
- Business continuity
- Termination of the contract

The extent of the services is the backbone of the SLA, as it essentially defines what is being sold; the performance measurement is then required to ensure that the agreed performance level is achieved. The problem resolution refers to the channel through which outages and other incidents are handled, and it’s closely linked with the security and business continuity of the cloud service. The cloud customer also has responsibilities, usually regarding access and security. The extent of these responsibilities is dependent on the chosen service model. Guarantees and remedies are relevant in situations where the quality of service is negatively affected; these define the compensation for such incidents. Defining security measures is perhaps the single most relevant point in the SLA regarding risks and their management in the cloud; they cover the essential security responsibilities, measures and procedures. The business continuity is an equally important part of the SLA, and it's tightly linked to the security topic. Finally, specific conditions such as cause and time frame for the termination of the contract are usually agreed on (Kandukuri et al. 2009, 518).

Clemons & Chen (2011, 8) list three key dimensions that need to be taken into account in cloud computing contracts: performance, security and legal recourse. The legal recourse defines how and by whom the legal claims need to be made, how those claims need to be backed and where the subsequent disputes will eventually be solved. Many big cloud service providers such as Amazon and Google put both the initiative and the burden of proof on the cloud customer. The provider doesn't take an active role; rather it's the customer who needs to present detailed evidence of the reduced performance or other issues (Patel, Ranabahu & Sheth 2009, 2–3). The legal recourse is an important part of the SLA, and it can be used to significantly reduce contractual risks; it is especially important if the cloud service provider and customer are located in different countries and thus within different jurisdictions. As such cases fall under the realm of international law, it is crucial to know which laws are applicable.

The second contractual dimension described by Clemons & Chen (2011, 8) is performance, which is a composition of multiple different metrics:

- Minimum service availability per given time period
- Minimum system response time
- Support response time
- Network and system stability
- Service quality and reliability

As can be seen from these metrics, the performance is a contract dimension that can easily be measured, and thus also easily enforced by the cloud customer. Some cloud customers might find it problematic that the cloud service provider is solely responsible for measuring the performance. To increase the reliability of the metrics, the task of measuring could be delegated to a third party to (Patel, Ranabahu & Sheth 2009, 2). The performance of the cloud is also strongly linked to the operational risks presented in chapter 4, specifically the availability. The uptime guaranteed in the SLA sets a baseline for cloud availability. Usually the uptime is defined as a percentage, i.e. the service is available, say 99,9% of the specified time frame, be it a week or a month. Usually it is also defined whether the guarantee applies to public network or a private network (Kandukuri, Paturi & Rakshit 2009, 519).

The third dimension presented by Clemons & Chen (2011, 8) is security. It is a broad and elemental area of cloud computing to be included in the SLA. It is very relevant to the cloud computing risks, specifically to data security. The ownership of and the access to the data need to be unambiguously defined in the SLA, so that there will be no uncertainties. Kalyvas, Overly & Karlyn (2013, 20–21) also discuss the definition of security in the context of the SLA, emphasizing the unique challenges posed to data security by the cloud as discussed earlier in chapter 4.2.

The SLA also has its limitations as a tool for managing outsourcing, specifically in the cloud realm. The traditional SLA is a static document written in natural language. Its conditions are based on the information available at the time of drafting and they are usually reviewed periodically or on demand. This process is manual and thus very slow, which limits the enforcement of the SLA in the cloud realm (Keller, Ludwig 2003, 58). Dillon, Wu & Chang (2010, 31) voice similar concerns, regarding the enforceability of the SLA in the dynamic cloud environment, where the constantly changing resource usage and the sheer number of different customers with their individual SLAs make the manual monitoring and management of the resources impossible. The self-service nature of the cloud requires automation for the resource provisioning, placing similar requirement for the SLA enforcement process.

To address this problem, a special adaptation of the SLA is proposed by Keller & Ludwig (2003, 58), namely the WSLA, which stands for Web Service Level Agreement. It is an adaption of the SLA, which takes into consideration the dynamic nature of web services; the model recognizes that different customer have different demands, which change dynamically. Patel et al. (2009, 3) have assessed the WSLA model and its compatibility to cloud computing. The model is divided into parties, SLA parameters and service level objectives, each with their own subsets:

- Parties
    - Service provider
    - Customer
    - Third party
- SLA parameters
    - Resource metrics
    - Composite metrics
    - Business metrics
- Service level objectives
    - If metric Y exceeds level X then action A
    - If metric Z goes under value R, then action B

The first two parties are the cloud service provider and the cloud customer. The third party could have various different roles depending on the service and deployment models. The third party could be performing a particular task, which is more specifically defined in the SLA parameters and objectives. Such a task could be for example the measuring of reaching availability levels defined in the SLA parameters and the subsequent actions caused by underperformance. The SLA parameters are metrics that are used to monitor the performance, usage and a multitude of other functions of the cloud. Resource metrics are derived straight from the service provider's unmodified data, example of such a metric is the transaction count. Composite metrics are a combination of multiple metrics, such as transactions per hour or day. The business metrics tie the SLA parameters to business goals individually for each customer (Patel, Ranabahu & Sheth 2009, 3).

Buyya, Yeo & Venugopal (2008, 4) argue that the rising popularity of cloud computing introduces inflexibility to the market, as the service providers are incapable of negotiating customer-specific SLAs due to the sheer number of cloud customers. This is a serious shortfall, as the details of the SLA are important to most customers. Many organizations have specific requirements, which have to be included in the SLA to ensure proper due diligence and the continuity of business operations despite incidents. Buyya et al. (2008, 5) propose to use service brokers, who are specialized in finding service providers with the given criteria and connect only those customers and providers who have similar expectations of the quality of service. This idea shares common ground with the WSLA model, as both models present an external party for handling the monitoring of the QoS. However, the model Buyya et al. propose is more market oriented whereas the WSLA model is built on the relationship between a single provider and customer.

Morin, Aubert & Gateau (2012, 5512–5513) propose an approach that combines the SLA with a risk management framework. The resulting service level framework would support real-time risk management and be adaptable to different architectures and service models; it would also be linked to operational management and modeling of risks. These goals can be achieved with the help of a policy-based approach combined with proper exception management to accommodate flexibility.

Kalyvas, Overly & Karlyn (2013, 20) present a concern regarding cloud computing involving multiple tiers, e.g. one party providing the service and another party hosting the data. In such cases, the third party should face the same requirements presented to the cloud service provider; otherwise the SLA would be pointless. In such cases some kind of an agreement should also be made with the third party; the original SLA alone is not sufficient to guarantee the level of service. More importantly, a separate agreement clarifies the legal obligations and makes the enforcement of the SLA more straightforward.

To conclude, the SLA is a critical document in cloud computing as it defines both the broad and the detailed terms of the relationship between the cloud customer and service provider. When there are clearly defined metrics for service measurement, responsibilities are unequivocally defined for both parties and potential legal issues are tied to certain legislation and jurisdiction, an organization has a firm grip of the security in the cloud.

## 6.2 Standards, frameworks & security models

The world of information systems is full of different standards; risk management in information systems is no different. Raz & Hillson (2005, 55) discuss nine different risk management standards, including national standards and standards defined by specialized organizations. The aim of the analysis was to compare various risk management standards on a regarding their scope, the analytical process and whether any of the standards have a specific emphasis. The following risk management standards were included in the analysis:

- IEEE Standard 1540-2001
- CEI/IEC 62198:2001
- JIS Q2001:2001(E)
- AS/NZS 4360:2004
- BS6079-3:2000
- CAN/CSA-Q850-97
- Risk Management Standard
- Project Risk Analysis & Management (PRAM) Guide
- Guide to the Project Management Body of Knowledge (PMBoK): Chapter 11, Project Risk Management

These frameworks were evaluated from various viewpoints, such as the organizing and planning of management of the risks; and the main phases of risk management: risk identification, risk analysis and risk treatment. The risk identification is elemental to any risk management process; identifying the relevant risks is a key part and starting point for proper risk management. In this study, the RQ1 is focused on the identification cloud-relevant risks. After risk identification, a further risk analysis is done to properly weigh the various risks in relation to their contexts. The focus in the RQ2 is on the evaluation of particular risks regarding their relevance to particular contexts, or in this case, the service models.

Raz & Hillson (2005, 58) recognize two distinct activities in the risk analysis phase, namely risk estimation and risk assessment. They define the risk estimation as the evaluation of the likelihood of the risks; the assessment is the process of prioritizing the estimated risks according to their treatment priority. The most common ways for risk treatment discovered in the analysis were risk avoidance, reducing the probability of risk, limiting the consequences of risks and finally transferring risks (Raz, Hillson 2005, 58). The conclusion of the analysis was that the standards were found to be similar in many regards, the biggest differences being in the planning phase, where the risk management principles, the organizational responsibilities and roles are defined.

Mather et al. (2009, 112–113) also discuss standards regarding cloud computing with a focus on security management in the cloud. They specifically discuss ITIL and ISO 27001/27002. ITIL is not a standard per se, but a framework gathering best practices in service management. Information security is an essential part of service management, and ITIL divides it into different categories:

- Policies
- Processes
- Procedures
- Work instructions

These components define the organization's objectives; the methods; and the responsibilities and specific steps needed for reaching these objectives. ITIL's focus on service management ties it closely with incident and change management, which have a big effect on the security in the cloud. ITIL is partly based on ISO/IEC 17799:2005 and ISO/IEC 20000. ISO 27001/27002, the other standard discussed by Mather et al. (2009, 113) consists of two parts. The ISO27001 is a certification standard, which strictly specifies the security management requirements that are needed for ISO-compliance, whereas ISO27002 sets guidelines to reach these requirements. The guidelines of ISO27002 can also be wholly or partially implemented in order to improve security management also without the ISO certification.

The division of the ISO 27001/27002 into two standards is more generally discussed by Borenstein & Blake (2011, 75). They divide standards into two distinct categories: prescriptive standards and evaluative standards. The former are detailed technical specifications and aim to cover all aspects of the technology whereas the evaluative standards gather best practices and processes and evaluate how well these are realized; they are also much less technical. The evaluative standards would provide a more valuable tool for companies considering the adoption of cloud computing. With a proper standard and framework the company could evaluate the service providers in order to find out which provider could be trusted  (Borenstein, Blake 2011, 76–77).

Borenstein et al. (2011, 75 – 76) also argue that the cloud computing paradigm doesn't create the need for any new prescriptive standards per se, as the majority of the technology utilized for cloud computing is already covered by different standards and is also well understood. The proprietary tools different cloud vendors have developed on their own further add complexity and unnecessary boundaries; it's also not very realistic to expect the providers to reveal the details of their proprietary technologies to their competitors. However, they recognize the need for evaluative standards. As the services are transferred over to the cloud and the company has significantly less power over them, it becomes crucial that there are tools in place to ensure the quality of the services.

Popović (2010, 346) discusses various standards, including the already discussed ITIL, ISO/IEC 20000 and ISO 27001/27002. They further present open virtualization format (OVF), which is an open standard specifically designed to improve the portability and platform independency of virtual systems. They argue that by utilizing open standards, organizations can introduce flexibility to deployment and utilize a broader range of providers due to the good compatibility, which in turn significantly decreases the risk of vendor lock-in. Another open solution, Eucalyptus is discussed by Nurmi, Wolski & Grzegorczyk (2009, 126). It’s a flexible architecture model based on modular components, it supports multiple web technologies including the Amazon’s EC2 and S3; it also provides a high level of isolation.

Cloud security alliance presents a collection of best practices in cloud security, gathered from vendors, customers and individuals with the aim of improving cloud security in general (Subashini, Kavitha 2011, 9). Vouk (2008, 242–243) presents concerns regarding the existing commercial solutions, which are often either hypervisor- or platform-specific and thus present an inherent risk of lock-in. They similarly propose the use of open solutions to improve the portability in the cloud. Beimborn, Miletzki & Wenzel (2011, 384) call for a general reference model, in order to mitigate this risk.

Kaliski Jr & Pauley (2010, 1–4), propose the use of thorough risk assessments as a tool to manage the risks in cloud computing. They further argue that the very properties that make the cloud so attractive to the cloud customers also make the proper assessment of privacy and security of the cloud very hard to conduct properly. The ultimate argument they make is that cloud services should be assessed by the same method they are provided; they propose risk assessment as a service. In the proposed model, the cloud security, availability and performance are monitored continuously on a real-time basis so that any disruptions, safety violations or security compromises can be acted upon immediately. Together with the more traditional periodical assessments, continuous checks form a risk management framework capable of dynamically adopting to the requirements posed by multitenancy and other ever-changing aspects of the cloud, thus providing reliable information of the state of the cloud.

To conclude, there are multiple non-profit organizations involved in developing cloud security in addition to the different national standards and standards created by specialized organizations. Some standards are more specific and require more effort from an organization, whereas some are broader and more adoptable to specific organizational needs; a good example of the latter is ISO 27002. Borenstein et al. (2011, 76) promote its adoption into wider use, Srinivasan (2012, 133) also propose to use the standard for benchmarking the security standards for the organization. The standards and frameworks are in themselves useful tools for benchmarking and mapping the capabilities of an organization and the preparedness against risks, but mapping and benchmarking alone are not sufficient to ensure risk resilience. The wide organizational adoption of the proposed measures into practice is an elemental part of risk management regardless of the framework or standard that is applied.

## 6.3 Trust

As discussed in chapter 4.4.1, trust is a key element in cloud computing relationships, and the lack thereof presents many risks to both parties. Pearson (2009, 6) argues that to improve trust, the cloud user's control over their data should be maximized and the provider's capabilities to access and process the data should be limited to the bare minimum. The control of the data refers to the knowledge of the whereabouts of the data and the ways in which the data is handled and processed. The more control over the data an organization has, the better their chances of maintaining its confidentiality and integrity. Khan & Malluhi (2010, 20) argue that distrust stems from the lack of transparency, fear of data loss and missing security assurances. Sangroya, Kumar, Dhok & Varma (2010, 261) note that the element of trust is absent from most cloud computing models, leading to perceived security weaknesses from the cloud customer's perspective. They divide trust into three distinct models: direct trust, transitive trust and assumptive trust; these were discussed earlier in chapter 4.4.1. Zhao et al. (2010, 190 – 192) expand on these trust models by proposing an approach, where trust is gained by introducing third parties into the equation. They suggest five different cloud deployment models, which differ in the responsibilities and the number of parties and the ways the services are set up:

- Separation model
- Availability model
- Migration model
- Tunnel model
- Cryptography model

The separation model adds a middleman into the traditional cloud equation; one provider is responsible for data manipulation; another provider is responsible for the data storage. This way none of the providers gains disproportionate control over the cloud customer. The availability model doubles on the separation model by adding another pair of providers: there are two providers doing the data manipulation and two providers providing the storage in a parallel set-up. The storage services are synchronized via a replication service in order to gain further redundancy. The third model, migration model is mainly based on a two-tier-architecture; however, in this model the data is replicated to a third party, differentiating the migration model from the separation model. The fourth model, tunnel model is similar to the separation model; the only difference is that the data flow between the two providers is encrypted. Finally the cryptography model adds a third party who is responsible for the encryption; otherwise the model is identical with the tunnel model (Zhao et al. 2010, 190–192). All these proposed models have the goal of mitigating the possibility of one party having a disproportionate control over the other. In order for the proposed deployment models to work, there is a need for high interoperability between different cloud services. These models are high-level concepts, and their implementation into practice requires a high level of interoperability, coordination and standardization; the technologies, such as cryptographic protocols and interfaces also need to be considered (Zhao et al. 2010, 193–193). Despite these obvious shortfalls, these models are a good starting point for finding solutions to the trust issues in the cloud.

Zissis & Lekkas (2012, 588) also present a model, where a trusted third party plays a major role. In their model, the third party reviews all transactions and communications between the cloud parties providing confidentiality, authentication and authorization. The third party is also responsible for ensuring the cryptographic protection and separation of the data. The advances of using the third party are the increased confidence and trust between parties; modern security technologies such as PKI (Public Key Infrastructure), SSO (Single Sign-On) and LDAP (Lightweight Directory Access Protocol) are utilized to ensure the CIA of the data in the cloud.

Chen, Paxson & Katz (2010, 5) discuss the prerequisites for the creation of trust in cloud computing; they promote mutual auditability as a tool that could be used to improve trust. Mutual auditability refers to audits performed by an independent third party on both the cloud customer and the vendor. The model challenges the traditional, logs- and records-based auditing process, where the documents to be audited are provided solely by the cloud service provider. Instead, both parties are placed under scrutiny; the ultimate aim of which is to ensure mutual trust. The mutual auditability relies on improved transparency, which is one approach to improving trust and thus the security in cloud computing. The open source initiatives presented in chapter 6.2 are a good example of transparency; as the technology is open to everyone, it can also freely be reviewed and analyzed as opposed to proprietary technology and standards. However, this transparency isn’t limited only to technology, but also to the relationship between the cloud customer and cloud service provider. Ramgovind et al. (2010, 6) propose greater transparency as the solution to security risk management in cloud computing. The more knowledge about the service provider, their operating procedures and security procedures is available, the easier it is to evaluate and adequately monitor the provided service. Ko et al. (2011, 1–2) list security, privacy, accountability and auditability as the trust components in the cloud; together they build they mitigate the barriers for cloud adoption and thus increase the confidence level the cloud customers place on the cloud.

Khan & Malluhi (2010, 21) divide trust into control, ownership, prevention and security. Control has a tendency to increase trust; the more control an organization has over the cloud where their data is stored, the more likely they also trust the cloud. The degree of ownership has a similar effect on trust. The added complexity in the cloud makes the ownership an important trust factor, as the ownership of the data might not always be clear when there are multiple parties involved. Prevention refers to the aim of the service provider to decrease the likelihood of adverse events; it is closely tied to the contractual aspects of the cloud. Khan & Malluhi note that to increase the level of trust, the focus should be on preventing trust violations rather than figuring out proper compensation for occurred trust violations; their argument is that money can't buy a new reputation in the case there is significant loss of customer data. The last aspect of trust, security, is an underlying topic throughout this study and is by nature a crucial part of trust (Khan & Malluhi 2010, 21).

Trust in the cloud is discussed also by Chow et al. (2009, 86–87), with the focus on third-party data control. They argue that the decreased control over data makes the tasks of auditing, due diligence and monitoring of contractual obligations more difficult to perform, in addition to increasing the risk of espionage, data lock-in and the transference of data to unknown parties. To avoid these risks, specifically the data lock-in and the possibility of the data ending up to the hands of malicious parties, trust-based cooperation between the different parties is prerequisite.

Khan & Malluhi (2010, 23–24) suggest encryption as a tool for enhancing the privacy and CIA of data, combined with emerging technologies such as remote access control, transparency-enabling tools and the proactive verification of the provider's reputation. Security certification and compartmentalization of the data into private enclaves to enable and ensure the isolation of the data are also suggested. Chow et al (2009, 89), develop the idea of using encryption further; they propose an approach called privacy-enhanced business intelligence. It's an approach that utilizes advanced cryptographic technologies to encrypt all the data in the cloud. The distinction to traditional encryption is that instead of having to decrypt the data to utilize it, queries and other computing operations are performed on the encrypted data, thus significantly increasing the overall data security. The approach is made possible by the advancements in cryptographic technologies; traditional encryption combined with database technologies doesn't provide the same capabilities.

Pearson & Benameur (2010, 696–670) divide trust into persistent trust and dynamic trust, persistent trust being the long-term trust in the infrastructure and its properties, the dynamic trust being a short-term, context-specific kind of trust. Further, they differentiate between technological trust and social trust; both aspects need to be considered; there is also the need to consider the trust relationships throughout the cloud supply chain. Third parties, in addition to providing additional trust in various set-ups, can also introduce uncertainties and liabilities that need to be recognized. These weak trust relationships often stem from the need to rapidly scale the cloud infrastructure according to demand.

The technological trust is touched upon also by Kumar et al. (2011, 416), who discuss the proceedings of the Trusted Computing Group (TCG). These proceedings include several points on the integrity of cloud computing systems, such as authenticating and asserting the origins of configuration changes and executables; verification of context for executing processes and tamper-proof, verifiable audit records. They also remind that there can never be complete trust in cloud computing; the technology dictates that there will always be lack of control and thus lack of trust inherent in the cloud.

In addition to the listed improvement areas, Kumar et al. (2011) propose the following ways to improve the level of trust in cloud computing:

- Development of the trust policies
- Trust semantics and assurance practices
- Privacy management technology
- Trusted identity management (IDM) solutions
- Development of interoperable, multi-platform key management systems

All these tools and technological approaches to trust improvement are only enablers of trust, not sources of trust themselves. Organizations need to consider their requirements regarding the degree of trust required and adopt the relevant tools to achieve this level; the sensitivity of the data is a big influencer on the trust level.

Finally, Vaquero et al. (2011, 11) argue that improving auditability is the key to the creation of trust, especially in case of complex trust chains created by multiple actors in the cloud supply chain. They promote an API-focused model, where the audit metrics are presented in a machine-readable form enabling the automation of the auditing process.

All in all, it is clear that to promote the wider adoption of cloud computing, the amount of trust needs to be sufficient. This is especially true when the cloud is used to store sensitive data that might be under strict regulations, for example the HIPAA-regulation that sets the limitations to the use of healthcare data of US citizens. Trust needs to be built on personal relations as well as technology capable of providing the tools for monitoring and enabling new kind of trust relationships emerging, making it possible for new kinds of services and business models to be built on top of cloud infrastructure. New standards and technologies enabling portability further enhance the security, confidentiality, integrity and availability of data and the proliferation of trust in the cloud, while decreasing the risk level of the cloud.

# 7 CONCLUSIONS

## 7.1 Conclusions

In this study, the various risks in cloud computing were discussed, based on the SPI-division of service models and the assumption of public cloud computing. The research question RQ1 focused on finding out what risks cloud computing presents to companies in general compared to the traditional, in-house IT infrastructure. Cloud computing, especially public cloud, is an inherently risky way of providing the computing needs for an organization, as the model always introduces additional actors and thus additional risks. The risks vary in their seriousness and relevance to different aspects of the cloud; there are some inherent technical risks in the cloud computing technology whereas the lack of trust and risks in outsourcing are social and legal in nature and thus need different risk management approaches.

There are risks to data security, which in the worst case might endanger the business operations of the whole company; these risks are related to data confidentiality, integrity and availability. The location and availability of the data also have significant implications to an organization, as they affect the regulations, rules and laws that apply; this is especially important when an organization uses sensitive data, such as health care information in their daily business operations.

Virtualization technology makes large-scale cloud computing possible and more importantly, viable by enabling the cloud service providers to utilize economies of scale in providing services at a negligible unit cost. This technology is complex, and in itself presents many risks that could compromise an organization's data. In cloud computing, the resources are accessed over the public Internet, which expands the trust boundaries of the This presents many threats that are not present in the traditional, isolated corporate networks, and dictates the need for secure web technologies and technologies for identity and access management.

The relationship between the cloud service provider and the cloud customer is another important factor to the risks in cloud computing. This relationship is founded on trust between the parties; trust is also a precondition to successfully adopting cloud computing. The lack of trust might lead to multiple problems such as vendor or data lock-in, effectively preventing the change of the service provider. There are also many risks that generally apply to the realm of outsourcing such as opportunity risks, reduced transparency and thus control and the risk of espionage.

The focus on RQ2 was on the relevance of the identified risks to each service model. Most of the risks in the cloud are relevant in all service models, specifically the risks that are on the lower end of the cloud stack. The higher-level risks, such as the web technology-related risks form an exception; they are not relevant in IaaS as the offerings are limited only to the OS-level.

Finally, RQ3 aimed to provide an answer to organizations could prepare themselves for these risks and manage, even mitigate them. Service level agreements, an elemental part of any outsourcing agreement, can be used as tool for managing the risks in cloud computing. The traditional, written SLAs might not suffice, however, which is why many new approaches embrace different technical solutions for monitoring and assessing the real-time performance and service level of the cloud. Different standards and frameworks provide a wide array of tools for evaluating and managing the risks in the cloud. Finally, trust both as a social construct and in the form of different trust models combining technology with third parties are a way of making cloud computing safer, more secure and most importantly, less risky.

Despite the delegation of many security responsibilities to the cloud service provider, in the end an organization is solely responsible for the security of their data in the cloud. Jansen & Grance conclude:

> *”Accountability for security and privacy in public cloud deployments cannot be delegated to a cloud provider and remains an obligation for the organization to fulfill.”(Jansen, Grance 2011, 52)*

## 7.2 Limitations and implications for future research

This study was conducted as a literature review, without empirical data. The aim was to gather knowledge about the risks in cloud computing in literature and to further analyze, which of these risks are commonly considered to be inherent to the cloud. As the pace of development in the cloud realm is very fast, studies about the most recent developments in the cloud were rather hard to find; two of the most recent studies reviewed for this study were from the years 2014 and 2015, respectively. A good example of the fast development is the introduction of container technology, which enables the cloud customers to move the data between clouds of different service providers, thus greatly increasing portability and decreasing the risk of lock-in. Unfortunately, there is little available research related to this area.

To further analyze the risk landscape of the cloud, the link between different service models and particular risks could be analyzed further as well as the effects of the deployment models to the risk. To conduct a more nuanced assessment of these relationships, empirical data would be preferential to another literature review.

## REFERENCES

Almorsy, M., Grundy, J. & Müller, I. (2010) An analysis of the cloud computing security problem. *Proceedings of APSEC 2010 Cloud Workshop, Sydney, Australia, 30th November.*

Andert, D., Wakefield, R. & Weise, J. (2002) Trust modeling for security architecture development, *Sun MicroSystems BluePrints.*

Barr, A. (2013) Amazon wins key cloud security clearance from government. Reuters. <http://www.reuters.com/article/2013/05/21/us-amazon-cloud-idUSBRE94K06S20130521>, retrieved 7.6.2013.

Beimborn, D., Miletzki, T. & Wenzel, S. (2011) Platform as a service (PaaS), *Business & Information Systems Engineering,* Vol. 3, no. 6, 381–384.

Bhardwaj, S., Jain, L. & Jain, S. (2010) Cloud computing: A study of infrastructure as a service (IAAS), *International Journal of engineering and information Technology,* Vol. 2, no. 1, 60–63.

Boniface, M., Nasser, B., Papay, J., Phillips, S.C., Servin, A., Yang, X., Zlatev, Z., Gogouvitis, S.V., Katsaros, G. & Konstanteli, K. (2010) Platform-as-a-service architecture for real-time quality of service management in clouds, *Internet and Web Applications and Services (ICIW), 2010 Fifth International Conference on*IEEE, 155.

Borenstein, N. & Blake, J. (2011) Cloud computing standards: Where's the beef?, *Internet Computing, IEEE,* vol. 15, no. 3, 74–78.

Brender, N. & Markov, I. (2013) Risk perception and risk management in cloud computing: Results from a case study of Swiss companies, *International Journal of Information Management,* Vol. 33, no. 5, 726–733.

Brian, O., Brunschwiler, T., Dill, H., Christ, H., Falsafi, B., Fischer, M., Grivas, S.G., Giovanoli, C., Gisi, R.E. & Gutmann, R. (2012) Cloud Computing, *White Paper SATW.*

Brown, W.J., Anderson, V. & Tan, Q. (2012) Multitenancy-security risks and countermeasures, *2012 15th International Conference on Network-Based Information Systems*IEEE, 7.

Chakraborty, R., Ramireddy, S., Raghu, T. & Rao, H.R. (2010) The information assurance practices of cloud computing vendors, *IT Professional Magazine,* Vol. 12, no. 4, 29.

Chen, D. & Zhao, H. (2012) Data security and privacy protection issues in cloud computing, *Computer Science and Electronics Engineering (ICCSEE), 2012 International Conference on*IEEE, 647.

Chow, R., Golle, P., Jakobsson, M., Shi, E., Staddon, J., Masuoka, R. & Molina, J. (2009) Controlling data in the cloud: outsourcing computation without outsourcing control, *Proceedings of the 2009 ACM workshop on Cloud computing security*ACM, 85.

Clemons, E.K. & Chen, Y. (2011) Making the decision to contract for cloud services: managing the risk of an extreme form of IT outsourcing, *System Sciences (HICSS), 2011 44th Hawaii International Conference on*IEEE, 1.

Columbus, L. (2016) Roundup Of Cloud Computing Forecasts And Market Estimates, 2016. Forbes <http://www.forbes.com/sites/louiscolumbus/2016/03/13/roundup-of-cloud-computing-forecasts-and-market-estimates-2016/#794f0c2074b0>, retrieved 18.6.2016

Dillon, T., Wu, C. & Chang, E. (2010) Cloud computing: Issues and challenges, *Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on*Ieee, 27.

Egwutuoha, I.P., Chen, S., Levy, D. & Calvo, R. (2013) Cost-effective Cloud Services for HPC in the Cloud: The IaaS or The HaaS?, *Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA)*The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 217.

ENISA, C.C. (2009) Benefits, risks and recommendations for information security, *European Network and Information Security.*

European Commission ruling (C131/12), (2014) Right to be forgotten

Fernandes, D.A., Soares, L.F., Gomes, J.V., Freire, M.M. & Inácio, P.R. (2014) Security issues in cloud environments: a survey, *International Journal of Information Security,* Vol. 13, no. 2, pp. 113–170.

Foster, I., Kesselman, C. & Tuecke, S. (2001) The anatomy of the grid: Enabling scalable virtual organizations, *International journal of high performance computing applications,* Vol. 15, no. 3, 200–222.

Foster, I., Zhao, Y., Raicu, I. & Lu, S. (2008) Cloud computing and grid computing 360-degree compared, *Grid Computing Environments Workshop, 2008. GCE'08*Ieee, 1.

Fox, A., Griffith, R., Joseph, A., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A. & Stoica, I. (2009) Above the clouds: A Berkeley view of cloud computing, *Dept.Electrical Eng.and Comput.Sciences, University of California, Berkeley, Rep.UCB/EECS,* Vol. 28.

Garfinkel, S.L. 2011, *The Cloud Imperative*, Technology Review, Inc., Cambridge, United States, Cambridge.

Garrison, G., Kim, S. & Wakefield, R.L. (2012) Success factors for deploying cloud computing, *Communications of the ACM,* Vol. 55, no. 9, 62–68.

Giessmann, A., Kyas, P., Tyrvainen, P. & Stanoevska, K. (2014) Towards a better Understanding of the Dynamics of Platform as a Service Business Models, *System Sciences (HICSS), 2014 47th Hawaii International Conference on*IEEE, 965.

Grobauer, B., Walloschek, T. & Stöcker, E. (2011) Understanding cloud computing vulnerabilities, *Security & privacy, IEEE,* Vol. 9, no. 2, 50–57.

Jansen, W. & Grance, T. (2011) Guidelines on security and privacy in public cloud computing, *NIST special publication,* Vol. 800, no. 144, 10–11.

Jensen, M., Schwenk, J., Gruschka, N. & Iacono, L.L. (2009) On technical security issues in cloud computing, *Cloud Computing, 2009. CLOUD'09. IEEE International Conference on*IEEE, 109.

Kaliski Jr, B.S. & Pauley, W. (2010) Toward Risk Assessment as a Service in Cloud Environments, *HotCloud*.

Kalyvas, J.R., Overly, M.R. & Karlyn, M.A. (2013) Cloud Computing: A Practical Framework for Managing Cloud Computing Risk--Part II, *Intellectual Property & Technology Law Journal,* Vol. 25, no. 4, 19–27.

Kandukuri, B.R., Paturi, V.R. & Rakshit, A. (2009) Cloud security issues, *Services Computing, 2009. SCC'09. IEEE International Conference on*IEEE, 517.

Kaplan, S. & Garrick, B.J. (1981) On the quantitative definition of risk, *Risk analysis,* Vol. 1, no. 1, 11–27.

Katzan Jr, H. (2010) On the privacy of cloud computing, *International Journal of Management and Information Systems,* Vol. 14, no. 2, 1.

Kaufman, L.M. 2009, "Data security in the world of cloud computing", *Security & Privacy, IEEE,* Vol. 7, no. 4, 61–64.

Keller, A. & Ludwig, H. (2003) The WSLA framework: Specifying and monitoring service level agreements for web services, *Journal of Network and Systems Management,* Vol. 11, no. 1, 57–81.

Khajeh-Hosseini, A., Greenwood, D. & Sommerville, I. (2010) Cloud migration: a case study of migrating an enterprise IT system to IaaS, *Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on*IEEE, 450.

Khan, K.M. & Malluhi, Q. (2010) Establishing trust in cloud computing, *IT professional,* Vol. 12, no. 5, 20–27.

Ko, R.K., Jagadpramana, P., Mowbray, M., Pearson, S., Kirchberg, M., Liang, Q. & Lee, B.S. (2011) TrustCloud: A framework for accountability and trust in cloud computing, *Services (SERVICES), 2011 IEEE World Congress on*IEEE, 584.

Krutz, R.L. & Vines, R.D. (2010) *Cloud security: A comprehensive guide to secure cloud computing,* Wiley Publishing, Indianapolis.

Kumar, P., Sehgal, V.K., Chauhan, D.S., Gupta, P. & Diwakar, M. (2011) Effective ways of secure, private and trusted cloud computing, *International Journal of Computer Science Issues,* Vol. 8, Issue 3, no. 2, 412.

Laplante, P.A., Zhang, J. & Voas, J. (2008) Distinguishing between Software Oriented Architecture and Software as a Service: What's in a Name?, *IEEE IT Professional,* Vol. 10, no. 3, 46–50.

Leitch, M. (2010) ISO 31000: 2009—the new international standard on risk management, *Risk Analysis,* Vol. 30, no. 6, 887–892.

Lenk, A., Klems, M., Nimis, J., Tai, S. & Sandholm, T. (2009) What's inside the Cloud? An architectural map of the Cloud landscape, *Proceedings of the 2009 ICSE Workshop on Software Engineering Challenges of Cloud Computing*IEEE Computer Society, 23.

Levy, Y. & Ellis, T.J. (2006) A systems approach to conduct an effective literature review in support of information systems research, *Informing Science: International Journal of an Emerging Transdiscipline,* Vol. 9, no. 1, 181–212.

Marston, S., Li, Z., Bandyopadhyay, S., Zhang, J. & Ghalsasi, A. (2011) Cloud computing—The business perspective, *Decision Support Systems,* Vol. 51, no. 1, 176–189.

Mather, T., Kumaraswamy, S. & Latif, S. (2009) *Cloud security and privacy: an enterprise perspective on risks and compliance,* O'Reilly.

Mell, P. & Grance, T. (2011) The NIST Definition of Cloud Computing.

Morin, J., Aubert, J. & Gateau, B. (2012) Towards cloud computing SLA risk management: issues and challenges, *System Science (HICSS), 2012 45th Hawaii International Conference on*IEEE, 5509.

Nurmi, D., Wolski, R., Grzegorczyk, C., Obertelli, G., Soman, S., Youseff, L. & Zagorodnov, D. (2009) The eucalyptus open-source cloud-computing system, *Cluster Computing and the Grid, 2009. CCGRID'09. 9th IEEE/ACM International Symposium on*IEEE, 124.

Okoli, C. & Schabram, K. (2010) A guide to conducting a systematic literature review of information systems research.

Patel, P., Ranabahu, A.H. & Sheth, A.P. (2009) Service level agreement in cloud computing.

Pearson, S. & Benameur, A. (2010) Privacy, security and trust issues arising from cloud computing, *Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference on*IEEE, 693.

Penzel, D., Kryvinska, N., Strauss, C. & Gregu, M. (2015) The Future of Cloud Computing: A SWOT Analysis and Predictions of Development, *Future Internet of Things and Cloud (FiCloud), 2015 3rd International Conference on*IEEE, 391.

Popović, K. (2010) Cloud computing security issues and challenges, *MIPRO, 2010 proceedings of the 33rd international convention*IEEE, 344.

Ramgovind, S., Eloff, M.M. & Smith, E. (2010) The management of security in cloud computing, *Information Security for South Africa (ISSA), 2010*IEEE, 1.

Raz, T. & Hillson, D. (2005) A comparative review of risk management standards, *Risk Management,* 53–66.

Rimal, B.P., Choi, E. & Lumb, I. (2009) A taxonomy and survey of cloud computing systems, *2009 Fifth International Joint Conference on INC, IMS and IDC*Ieee, 44.

Ristenpart, T., Tromer, E., Shacham, H. & Savage, S. (2009) Hey, you, get off of my cloud: exploring information leakage in third-party compute clouds, *Proceedings of the 16th ACM conference on Computer and communications security*ACM, 199.

Robinson, N., Valeri, L., Cave, J., Starkey, T., Graux, H., Creese, S. & Hopkins, P.P. (2010) The cloud: understanding the security, privacy and trust challenges, *Privacy and Trust Challenges (November 30, 2010).*

Rousseau, D.M., Manning, J. & Denyer, D. (2008) 11 Evidence in Management and Organizational Science: Assembling the Field's Full Weight of Scientific Knowledge Through Syntheses, *The academy of management annals,* Vol. 2, no. 1, 475–515.

Sangroya, A., Kumar, S., Dhok, J. & Varma, V. (2010) Towards analyzing data security risks in cloud computing environments, *Information Systems, Technology and Management* Springer, 255–265.

Smith, A.D. & Rupp, W.T. (2002) Application service providers (ASP): moving downstream to enhance competitive advantage, *Information Management & Computer Security,* Vol. 10, no. 2, 64–72.

Stoneburner, G., Goguen, A. & Feringa, A. (2002) Risk management guide for information technology systems, *Nist special publication,* Vol. 800, no. 30, 800–830.

Subashini, S. & Kavitha, V. (2011) A survey on security issues in service delivery models of cloud computing, *Journal of Network and Computer Applications,* Vol. 34, no. 1, 1–11.

Sultan, N.A. (2011) Reaching for the "cloud": How SMEs can manage, *International Journal of Information Management,* Vol. 31, no. 3, 272–278.

Torraco, R.J. (2005) Writing integrative literature reviews: Guidelines and examples, *Human Resource Development Review,* Vol. 4, no. 3, 356–367.

Tranfield, D., Denyer, D. & Smart, P. (2003) Towards a methodology for developing evidence-informed management knowledge by means of systematic review, *British Journal of Management,* Vol. 14, no. 3, 207–222.

Tsai, H., Siebenhaar, M., Miede, A., Huang, Y. & Steinmetz, R. (2012) Threat as a Service: Virtualization's Impact on Cloud Security, *IT Professional Magazine,* Vol. 14, no. 1, 32.

Turner, M., Budgen, D. & Brereton, P. (2003) Turning software into a service, *Computer,* Vol. 36, no. 10, 38–44.

Vaquero, L.M., Rodero-Merino, L., Caceres, J. & Lindner, M. (2008) A break in the clouds: towards a cloud definition, *ACM SIGCOMM Computer Communication Review,* Vol. 39, no. 1, 50–55.

Vaquero, L.M., Rodero-Merino, L. & Morán, D. (2011) Locking the sky: a survey on IaaS cloud security, *Computing,* Vol. 91, no. 1, 93–118.

Webster, J. & Watson, R.T. (2002) Analyzing the past to prepare for the future: Writing a literature review, *Management Information Systems Quarterly,* vol. 26, no. 2, 3.

Venters, W. & Whitley, E.A. (2012) A critical review of cloud computing: researching desires and realities, *Journal of Information Technology,* Vol. 27, no. 3, 179–197.

Vogels, W. (2009) Eventually consistent, *Communications of the ACM,* Vol. 52, no. 1, 40–44.

Von Solms, R. & Van Niekerk, J. (2013) From information security to cyber security, *Computers & Security,* Vol. 38, 97–102.

Vouk, M.A. (2008) Cloud computing—Issues, research and implementations, *Information Technology Interfaces, ITI 2008, 30th International Conference on*Ieee, 31.

Wu, W., Lan, L.W. & Lee, Y. (2011) Exploring decisive factors affecting an organization's SaaS adoption: A case study, *International Journal of Information Management,* Vol. 31, no. 6, 556–563.

Youseff, L., Butrico, M. & Da Silva, D. (2008) Toward a unified ontology of cloud computing, *Grid Computing Environments Workshop, GCE'08*IEEE, 1.

Yu, X. & Wen, Q. (2010) A view about cloud data security from data life cycle, *Computational Intelligence and Software Engineering (CiSE), 2010 International Conference on*IEEE, 1.

Zhao, G., Rong, C., Jaatun, M.G. & Sandnes, F.E. (2010) Deployment models: Towards eliminating security concerns from cloud computing, *High Performance Computing and Simulation (HPCS), 2010 International Conference on*IEEE, 189.

Zhou, M., Zhang, R., Xie, W., Qian, W. & Zhou, A. (2010) Security and privacy in cloud computing: A survey, *Semantics Knowledge and Grid (SKG), 2010 Sixth International Conference on*IEEE, 105.

Zissis, D. & Lekkas, D. (2012) Addressing cloud computing security issues, *Future Generation Computer Systems,* Vol. 28, no. 3, 583–592.