Cloud Encryption and Key Management Considerations

Daniel Cuschieri
March 2013

Submitted as part of the requirements for the award of the
Master of Science in Information Security

Royal Holloway, University of London
Egham, Surrey TW20 0EX, England
Cloud Encryption and Key Management Considerations

Specially dedicated to
my beloved family
and close friends
Acknowledgments

This work would not have been possible without the support and encouragement of my supervisor, Dr Po Wah Yau. My most grateful appreciation goes to him, for his patience and support through the entire supervision of this dissertation, and for taking time to answer queries related to this project. His enthusiasm on the subject was essential for successfully completing this dissertation.

Last, but certainly not the least, I would also like to thank my family, especially my parents, on whose constant encouragement I have relied. Their support and encouragement was fundamental for completing the project, and it is to them that I dedicate this work.
## Table of Contents

1  Introduction .................................................................................................................. 1
   1.1  Objectives ............................................................................................................. 1
   1.2  Methodology and Structure ................................................................................. 2

2  Cloud Computing .......................................................................................................... 4
   2.1  Definition .............................................................................................................. 4
   2.2  Characteristics ..................................................................................................... 6
   2.3  Service Models ..................................................................................................... 7
       2.3.1  Software as a Service (SaaS) ....................................................................... 7
       2.3.2  Platform as a Service (PaaS) ...................................................................... 8
       2.3.3  Infrastructure as a Service (IaaS) ................................................................. 9
   2.4  Deployment Models ............................................................................................... 9
   2.5  Taxonomy of Cloud Computing Use Cases ......................................................... 10
   2.6  Conceptual Models ............................................................................................ 12
       2.6.1  Model A – Public SaaS .............................................................................. 13
       2.6.2  Model B – Hybrid SaaS ............................................................................. 13
       2.6.3  Model C – Public PaaS .............................................................................. 14
       2.6.4  Model D – Private IaaS ............................................................................. 15
       2.6.5  Model E – Public IaaS .............................................................................. 15
       2.6.6  Summary ..................................................................................................... 16

2.7  Benefits and Concerns of Cloud Computing ......................................................... 16
2.8  Conclusion ................................................................................................................ 18

3  Cloud Security .............................................................................................................. 20
   3.1  Organisations Working on Cloud Security ......................................................... 20
   3.2  Cloud Security Threats ....................................................................................... 22
       3.2.1  Virtually Unlimited Compute Power ............................................................... 22
       3.2.2  Cloud Management ...................................................................................... 23
       3.2.3  Cloud Provider Insiders .............................................................................. 23
       3.2.4  Shared Cloud Infrastructure ....................................................................... 24
       3.2.5  Data Security ............................................................................................... 24
       3.2.6  Access to Cloud Data .................................................................................. 25
       3.2.7  Unknown Risk Profile ................................................................................ 26
       3.2.8  Forensics ..................................................................................................... 27
   3.3  Perception of Cloud Security .............................................................................. 27
       3.3.1  Survey Methodology ................................................................................... 27
       3.3.2  Survey Results ............................................................................................. 28
   3.4  Cloud Computing Risk Assessment Method ..................................................... 31
       3.4.1  CC-RAM Design Decisions ........................................................................ 34

3.5  Conclusion ................................................................................................................ 36

4  Encryption and Key Management in the Cloud ......................................................... 38
   4.1  Cryptography Services Supporting the Cloud .................................................... 38
   4.2  Encryption as a Threat Countermeasure ............................................................... 39
   4.3  Where to Encrypt Data ....................................................................................... 42
   4.4  Encryption in Different Cloud Architectures ....................................................... 45
       4.4.1  Encryption in SaaS ...................................................................................... 45
       4.4.2  Encryption in PaaS ....................................................................................... 47
       4.4.3  Encryption in IaaS ...................................................................................... 48
   4.5  Cloud Encryption Trends ....................................................................................... 50
   4.6  Conclusion ............................................................................................................ 51

5  Challenges of Cloud Encryption ................................................................................. 53
Cloud Encryption and Key Management Considerations

5.1 Encryption Challenges ........................................................................................................... 53
5.1.1 Inherent Challenges ........................................................................................................... 53
5.1.2 Challenges in Implementing Desirable Functionality ......................................................... 54
5.1.3 Cloud Model Analysis ......................................................................................................... 55
5.2 Key Management Challenges ............................................................................................... 57
5.2.1 Inherent Challenges ........................................................................................................... 58
5.2.2 Cloud Model Analysis ......................................................................................................... 59
5.3 Legislation and Compliance Challenges ................................................................................... 61
5.3.1 Legislation Which Affects Cloud Encryption ...................................................................... 61
5.4 Information Security Standards ............................................................................................... 63
5.4.1 Generic Information Security Standards ............................................................................ 64
5.4.2 Cloud Security Standards ................................................................................................. 66
5.5 Conclusion ............................................................................................................................... 68
6 Best Practices and Proposals for Cloud Key Management ....................................................... 69
6.1 Use of Key Management Standards ....................................................................................... 69
6.2 Key Generation and Storage ..................................................................................................... 71
6.2.1 Split Key Management ....................................................................................................... 72
6.3 Key Availability ....................................................................................................................... 73
6.4 Key Distribution ....................................................................................................................... 74
6.4.1 Federated Key Management ............................................................................................... 74
6.5 Key Termination ........................................................................................................................ 76
6.6 Key Management Policy .......................................................................................................... 76
6.7 Separation of Duties .................................................................................................................. 76
6.8 Key Management Interoperability ........................................................................................... 78
6.8.1 OASIS Key Management Interoperability Protocol ......................................................... 78
6.9 Human Aspect of Key Management ....................................................................................... 84
6.10 Conclusion ............................................................................................................................... 84
7 Best Practices and Proposals for Cloud Data Encryption ......................................................... 85
7.1 Encryption Policy ...................................................................................................................... 85
7.2 Encryption Algorithms ............................................................................................................. 86
7.3 Integrity of Encrypted Data ...................................................................................................... 87
7.4 Encryption Client Security ....................................................................................................... 88
7.5 Legislation and Compliance ..................................................................................................... 88
7.6 Encryption that Enables Easier Management .......................................................................... 89
7.6.1 Identity Based Encryption ................................................................................................. 89
7.6.2 Attribute Based Encryption .............................................................................................. 90
7.7 Encryption Interoperability ...................................................................................................... 92
7.8 Implementation of Desirable Features .................................................................................... 92
7.8.1 Processing Encrypted Data ............................................................................................... 93
7.8.2 Searching and Sorting Encrypted Data ............................................................................. 95
7.9 Conclusion ............................................................................................................................... 97
8 Dropbox – A Cloud Encryption Case Study .............................................................................. 99
8.1 Cloud Classification .................................................................................................................. 99
8.2 CC-RAM Analysis of Dropbox ............................................................................................... 100
8.2.1 Asset Identification .......................................................................................................... 100
8.2.2 Asset Valuation ................................................................................................................. 100
8.2.3 Deployment Model Analysis ............................................................................................. 101
8.2.4 Service Model Analysis ................................................................................................... 101
8.2.5 Data Flow Identification .................................................................................................. 101
8.2.6 Threat and Vulnerability Analysis .................................................................................... 101
8.2.7 Risk Score Calculation and Risk Treatment Planning ..................................................... 103
8.2.8 Countermeasure Identification and Gap Analysis ........................................................... 104
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.2.9</td>
<td>Feedback Loops</td>
<td>106</td>
</tr>
<tr>
<td>8.3</td>
<td>Security of Dropbox Competitors</td>
<td>107</td>
</tr>
<tr>
<td>8.4</td>
<td>Conclusion</td>
<td>107</td>
</tr>
<tr>
<td>9</td>
<td>Conclusion</td>
<td>109</td>
</tr>
<tr>
<td>9.1</td>
<td>First Objective</td>
<td>109</td>
</tr>
<tr>
<td>9.2</td>
<td>Second Objective</td>
<td>112</td>
</tr>
<tr>
<td>9.3</td>
<td>Third Objective</td>
<td>115</td>
</tr>
<tr>
<td>9.4</td>
<td>Future Work and Concluding Remarks</td>
<td>117</td>
</tr>
<tr>
<td>Bibliography</td>
<td></td>
<td>118</td>
</tr>
<tr>
<td>Appendix A – Security Effectiveness Score</td>
<td>125</td>
<td></td>
</tr>
</tbody>
</table>
# Table of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1</td>
<td>Visual Model of NIST Working Definition of Cloud Computing</td>
<td>5</td>
</tr>
<tr>
<td>Figure 2</td>
<td>Cloud Computing Use Cases Taxonomy</td>
<td>11</td>
</tr>
<tr>
<td>Figure 3</td>
<td>The Main Security Concerns of Cloud Computing</td>
<td>29</td>
</tr>
<tr>
<td>Figure 4</td>
<td>Relationship between Security Posture and Cloud Adoption</td>
<td>30</td>
</tr>
<tr>
<td>Figure 5</td>
<td>Cloud Computing Risk Assessment Method (CC-RAM)</td>
<td>32</td>
</tr>
<tr>
<td>Figure 6</td>
<td>Security and User Control in Cloud Service Models</td>
<td>45</td>
</tr>
<tr>
<td>Figure 7</td>
<td>Where Cloud Encryption is Applied</td>
<td>50</td>
</tr>
<tr>
<td>Figure 8</td>
<td>Who Manages Cloud Encryption Keys</td>
<td>51</td>
</tr>
<tr>
<td>Figure 9</td>
<td>Plan Do Check Act Cycle of ISO27001</td>
<td>64</td>
</tr>
<tr>
<td>Figure 10</td>
<td>OSA Cloud Computing Pattern</td>
<td>67</td>
</tr>
<tr>
<td>Figure 11</td>
<td>Split Key Management</td>
<td>73</td>
</tr>
<tr>
<td>Figure 12</td>
<td>Cloud Key Management Service</td>
<td>77</td>
</tr>
<tr>
<td>Figure 13</td>
<td>Remote Key Management Service</td>
<td>77</td>
</tr>
<tr>
<td>Figure 14</td>
<td>Client Side Key Management</td>
<td>78</td>
</tr>
<tr>
<td>Figure 15</td>
<td>Proliferation of Key Management Systems</td>
<td>79</td>
</tr>
<tr>
<td>Figure 16</td>
<td>Key Management with KMIP</td>
<td>79</td>
</tr>
<tr>
<td>Figure 17</td>
<td>KMIP Operations, Objects and Attributes</td>
<td>80</td>
</tr>
<tr>
<td>Figure 18</td>
<td>KMIP KMS at Cloud Consumer, Encryption System at Cloud Provider</td>
<td>82</td>
</tr>
<tr>
<td>Figure 19</td>
<td>KMIP KMS and Encryption System at Cloud Provider</td>
<td>82</td>
</tr>
<tr>
<td>Figure 20</td>
<td>KMIP KMS at Trusted Third Party, Encryption System at Cloud Provider</td>
<td>83</td>
</tr>
<tr>
<td>Figure 21</td>
<td>Dropbox Cloud Synchronisation</td>
<td>99</td>
</tr>
<tr>
<td>Figure 22</td>
<td>Risk Matrix</td>
<td>104</td>
</tr>
</tbody>
</table>
Executive Summary

In recent years, there has been a surge in interest in Cloud Computing, which involves various hardware, software and other existing technologies combined together to provide consumers with services that are accessible over a network. Apart from reducing costs, the Cloud increases scalability and introduces significant flexibility, since this infrastructure can be configured in a wide variety of ways depending upon the user’s requirements. At the two extremes, an organisation that is moving to the Cloud can choose between private and public Clouds, which introduce more security and scalability, respectively. Moreover, one can choose between Software as a Service, Platform as a Service or Infrastructure as a Service, with each service model being particularly suited in different scenarios.

Despite being an interesting prospect for organisations, the Cloud introduces a number of concerns which mainly revolve around security considerations. It is, therefore, essential that these risks are identified and addressed, in order to ensure an adequate level of security. This can be done through a comprehensive and methodical Cloud risk assessment, which is typically a more complex process compared to that of traditional IT systems, owing to the inherent characteristics of the Cloud. In recent years, frameworks and models have been put forward, which cover different parts of a Cloud risk assessment. Based upon these, a Cloud Computing Risk Assessment Method (CC-RAM) is proposed in this study, which is a complete process for analysing and evaluating the Cloud’s risks. This cyclic model consists of eleven main phases, which cover all processes of a traditional risk assessment, interspersed with Cloud-specific analysis phases.

As part of the Cloud risk assessment, one needs to analyse the threats involved. Besides the generic information security threats of confidentiality, integrity and availability, Cloud-specific threats should also be considered. The threat of an attacker having virtually unlimited Cloud compute power, as well as the threat of attacks on Cloud management interfaces should both be considered. Cloud provider insiders and the fact that the Cloud infrastructure is shared by different tenants, are also a cause of concern. Moreover, threats to the data’s security and to its access control mechanisms should also be considered during a Cloud threat assessment. The Cloud also provides the threat of having an unknown risk profile and the lack of sufficient forensic and auditing information. The susceptibility of the Cloud to these threats depends upon many factors, including the service and deployment models used.

Following a risk assessment, various countermeasures need to be implemented to address the threats and mitigate the risks involved. Encryption is one such measure, which can be used to address risks related to Cloud data confidentiality. This can be applied to data that is at rest, in transit or in use, the choice of which depends upon the risk assessment findings. However, despite its clear advantages, encryption is challenging to implement and manage. Amongst the main challenges presented, one finds the management of encryption policies and the availability and integrity of encrypted data. Other challenges involve ensuring security of the encryption client, as well as compliance with legislation and standards. Apart from these inherent challenges, when performing Cloud encryption, one should also attempt to implement desirable functionality such as searching and sorting of encrypted data. The processing of data directly in its encrypted form, as well as the use of interoperable encryption algorithms are also desirable but challenging tasks. Moreover, when considering encryption, one needs to also understand and address the challenges associated with its corresponding key management. These include key generation, storage, disposal and expiration, which should be governed by a suitable key management policy. Interoperability and availability of the key management function should also be addressed, and the principle of separation of duties must be enforced.
Even though it is a challenge to implement and manage encryption, its merits should not be ignored since it can help increase information security in the Cloud. Encryption, in fact, plays an important role not only in implementing generic information security standards such as ISO27001, but also Cloud-specific standards such as the Cloud Computing Pattern. By following best practices in this field, the process of implementing Cloud encryption and key management can be simplified and improved. A set of such best practices covering various aspects of these two processes are given in this study, together with a discussion on their applicability in different Cloud models. A number of technologies which aim to implement the suggested best practices are also proposed, to demonstrate how these can be achieved through the use of established or trending technologies, and current research. It is hoped that such best practices can help reduce the challenges associated with Cloud encryption, in the process increasing Cloud security and further encouraging its uptake.
1 Introduction

The Cloud involves a collection of hardware, networks, storage and services that are managed by a provider, and made available to the end consumers as a service. In recent years, interest in the Cloud for both personal and business use has shown an upwards trend, due to the flexibility that this provides. Cloud Computing can be used to reduce the constraints of traditional computing, benefiting the company in terms of space, time, power, cost and simplified business processes. However, just like with any computing system, when considering the Cloud one needs to also evaluate the close links that this has to information security. Security, in fact, is a major consideration in Cloud Computing, since when the organisation moves its data and operations onto the Cloud, it essentially loses much of the control that it previously had. Various techniques can be used to mitigate these security risks, including technical and procedural controls which can help provide some reassurance of confidentiality, integrity and availability to the Cloud users. Encryption and key management are two controls that are often used to achieve information security in traditional computing environments. In this study, their applicability in a Cloud environment will be analysed, in order to understand whether such measures can help in achieving information security in the Cloud, thereby further encouraging its adoption.

1.1 Objectives

The aim of this study is to understand what the Cloud’s security considerations are. The suitability of encryption and key management to theoretically address some of these issues, at least in part, will also be evaluated. Moreover, this study will investigate the current state of affairs in Cloud encryption, in order to determine where encryption is falling short from addressing these Cloud security issues, and why. Key management, which is strongly related to encryption, will also be considered.

More specifically, three objectives are set in this study, in order to be able to arrive to such an analysis.

To define Cloud Computing, and to understand the security concerns involved
Cloud Computing is considered to be quite a new phenomenon in IT, which has gained popularity in recent years – a trend which is constantly on the rise. However, security considerations are the main factors that are limiting its uptake in certain domains. It is, therefore, important to understand what the security concerns are, both from a users’ perspective and through a more formal security assessment model.

To understand the role of encryption in the Cloud, and to outline challenges faced in its implementation
Once the security concerns have been identified, it is then possible to analyse what role encryption and good key management can play in achieving Cloud security. Whilst encryption might be suitable for addressing certain security concerns, this is however not a panacea that solves all security issues. When implementing security controls such as encryption and key management, one must first consider what challenges will be faced in the process. Apart from technical and procedural issues, legal and compliance concerns also need to be evaluated, something which is complicated further due to the geographically-dispersed nature of the Cloud. A set of best practices related to Cloud Computing encryption and key management will be derived in this study.
To review encryption and key management approaches that aim to address the Cloud security issues

Different Cloud infrastructures adopt diverse approaches to encryption. Some opt to perform client-side encryption, whilst other providers use server-side encryption. Other Cloud applications simply do not have any inbuilt data encryption functionality, in which case it might be possible to use third party systems to provide encryption services for the Cloud data. Encryption and key management are complex processes that need to be considered carefully, especially if they are to be secure within a complex environment such as the Cloud. Moreover, it is important that encryption, like any other security measures adopted, is not too limiting for the end user to actually use as this would otherwise be unsuitable. Various approaches and techniques for key management and encryption exist, some of which are very recent additions to Cloud Computing. Whilst a comprehensive review of all approaches to encryption is beyond the scope of this study, some of these techniques will be analysed, in order to determine how and why they can improve Cloud Computing, and what their limitations are.

1.2 Methodology and Structure

This study consists of a significant degree of research, as well as independent work and analysis done by the author. This study aims to analyse the problem of Cloud confidentiality in order to understand the advantages and challenges involved in achieving this. A number of methodologies and best practices that should be employed to achieve this confidentiality in the Cloud, are also analysed to determine how they help solve the problem at hand. In recent years, Cloud Computing has been discussed by various experts, and this study is hence based upon the available literature. However, the author also proposes various models, classifications and best practices in this study. The report is structured into a number of Sections which help achieve a comprehensive and cohesive analysis that helps explain and analyse the problem. It is hoped that the insight given in this study will be useful to others who opt to carry out further research in this field.

In Chapter 2, “Cloud Computing”, the concept of Cloud Computing is introduced by reviewing various definitions that have been put forward by experts in this area. The essential characteristics of Cloud Computing are also reviewed, together with the various service and deployment models that exist. A taxonomy for the Cloud’s use cases is proposed in this Chapter, in order to structure the several types of Clouds into a more manageable way. Conceptual models based upon typical configurations of the Cloud, are also defined in this Chapter, to guide further analysis on Cloud information security in this study. The benefits and concerns of the Cloud are also introduced in this Chapter, since they justify the need for research in the field of Cloud Computing.

Chapter 3, “Cloud Security”, proceeds to analyse in more detail the security considerations of Cloud Computing. Any research on information security has, at its foundations, a threat and risk assessment. In this study, Cloud-specific information security threats are defined, and these are then analysed in the light of the conceptual models proposed earlier in this study. Moreover, this Chapter also analyses what the perception on Cloud security is, since perceptions are arguably just as important as actual security itself. Surveys carried out by reputable sources are used to guide this discussion. Finally, the chapter ends with the proposal of a Cloud Computing Risk Assessment Model, which expands the traditional risk assessment process to include Cloud-specific considerations. This model is then referenced in subsequent analysis done in this study.

Chapter 4, “Encryption and Key Management in the Cloud”, starts by introducing the use of cryptography in general as a suitable technique for supporting security in Cloud implementations.
More specifically, the Chapter proceeds to analyse encryption as a possible measure for addressing the Cloud threats identified earlier in the study. This investigation delves deeper into the topic by also reviewing the suitability of encryption in the various Cloud service models. The Chapter concludes by analysing trends in the usage of encryption in Cloud Computing, through the use of reliable surveys.

In Chapter 5, “Challenges of Cloud Encryption”, a more in-depth analysis is done on the challenges involved when implementing encryption in a Cloud environment. These challenges can be split into those that are inherent to Cloud encryption, and those which are encountered when attempting to implement non-essential but highly desirable encryption functionality. The conceptual models defined earlier in the study are used to guide the discussion and analysis of this subject. When implementing Cloud encryption, one also needs to consider key management since this is an essential component of any cryptosystem. For this reason, this Chapter also reviews the inherent challenges of key management, and provides a discussion on these challenges through the use of the conceptual models. Challenges related to legislation and compliance when using Cloud encryption are also reviewed. A discussion on how Cloud-specific standards as well as generic information security standards can be applied to the Cloud in order to address these challenges is also done in this Chapter.

Possible approaches to key management and encryption are discussed in Chapter 6, “Best Practices and Proposals for Cloud Key Management” and Chapter 7, “Best Practices and Proposals for Cloud Data Encryption”, respectively. These chapters discuss several best practices which can be used to counter the various threats and challenges described throughout prior chapters of the study. Moreover, these Chapters also propose the use of emerging technologies to counter these threats where established best practices do not suffice. Although not being necessarily sufficiently refined or established to be currently adopted as best practices, these technologies might be a step in the right direction for addressing these threats in the near future, and they are hence described and analysed in these Chapters. The different Cloud architectures as well as the conceptual models described earlier in the study are referenced when discussing the relevance of these best practices and proposals for achieving Cloud security.

Chapter 8, “Dropbox – A Cloud Encryption Case Study”, attempts to summarise this research by applying the key concepts of this study to a real-world scenario. An overview of Dropbox is given, as well as its classification according to the taxonomy defined earlier in the study. The Cloud Computing Risk Assessment Method is then applied to Dropbox, identifying the threats involved. Suitable countermeasures are identified and a gap analysis is also performed in this Chapter.
2 Cloud Computing

The term Cloud Computing, which is a relatively recent addition to the Information Technology (IT) vocabulary, is used to encompass a number of concepts which have been established before the name was even coined. Since its introduction, this has drawn interest and attention from different participants, such as academics, businesses and organisations. Various definitions of Cloud Computing have been put forward, most notably the one by the Cloud Security Alliance, which looks at the Cloud from a security perspective. This definition is built upon the working definition provided by the US National Institute of Standards and Technology, which describes the Cloud in terms of five characteristics, three service models, and four deployment models. In Section 2.1 of this study, the term Cloud Computing is defined from a security perspective, whilst Section 2.2 builds upon this definition by introducing various characteristics that are essential to the Cloud. The service models and the deployment models are discussed in Section 2.3 and Section 2.4, respectively.

Due to the many different flavours and models that the Cloud comes in, the Cloud’s uses are also vast. In order to make it easier to evaluate the advantages and disadvantages of the Cloud, a taxonomy can be used to categorise the various use cases into a manageable classification. In Section 2.5, a possible taxonomy for Cloud use cases is proposed. A set of conceptual models based upon different combinations of service and deployment models will be provided in Section 2.6, and these will be referenced throughout the rest of the study when analysing different approaches to Cloud information security.

As with any model or technology, the Cloud brings about with it a number of benefits as well as issues or concerns, which need to be thoroughly understood and evaluated when considering a Cloud Computing implementation. This aspect is discussed in Section 2.7, which covers the pros and cons of a Cloud environment.

2.1 Definition

Although the term Cloud Computing (or simply the Cloud) is now widely used by many researchers and organisations, the definitions that they put forward often differ substantially. It is, therefore, appropriate to understand what the Cloud stands for, seeing that it incorporates a variety of technologies and approaches. One of the most popular definitions is that provided by Gartner, where the Cloud is defined as “a style of computing where massively scalable and elastic IT-related capabilities are provided ‘as a service’ using Internet Technologies to multiple external customers” (Ilyenga, 2009). The Cloud has also been described as “hosted online services [that] are accessed via the internet, which is metaphorically depicted as a ‘Cloud’. Usually, the graphical user interface is provided by the customer’s own web browser” (KPMG Netherlands, 2010). Whilst both of these definitions are correct, the emphasis on the importance of data in Cloud Computing is best described by Bruce Schneier when he states that “Cloud Computing is storing your data on someone else’s computer and accessing it via a network” (Chung & Hermans, 2010).

As can be seen from the above definitions, the Cloud has many facets making it one of the most complex and sophisticated, yet appealing, models. The Cloud Security Alliance, which is a non-profit organisation which aims to promote security within Cloud Computing, provides yet another attempt at defining this term, this time from a security professional’s point of view:
Cloud Computing is “an evolving term that describes the development of many existing technologies and approaches to computing into something different. Cloud separates application and information resources from the underlying infrastructure, and the mechanisms used to deliver them”. Moreover, it “enhances collaboration, agility, scaling, and availability, and provides the potential for cost reduction through optimized and efficient computing... Cloud describes the use of a collection of services, applications, information, and infrastructure comprised of pools of compute, network, information, and storage resources” that can be acquired or released as necessary.

**Definition 1 - Cloud Computing**  
Source: (Cloud Security Alliance, 2009)

Seeing that this definition is sufficiently comprehensive and it considers Cloud Computing from a security perspective, this will be the one adopted throughout this study.

Having identified a suitable definition for Cloud Computing, one also needs to consider whether the Cloud is something theoretical, or if there is demand for such a model in today’s IT environment. With advances in technology, such as faster and larger servers, more reliable networks and increased internet bandwidth, there has been a surge in interest in the concept of Cloud Computing. In fact, trends indicate that organisations are increasingly recognising the importance of the Cloud and regard this as the future of IT rather than simply a temporary hype. In a recent study, KPMG found that 83% of organisations that were involved in the survey recognise the significance of the Cloud, with 49% of respondents also considering the Cloud as an interesting technical or business offering (KPMG Netherlands, 2010).

Due to this increase in popularity of the Cloud, the US National Institute of Standards and Technology (NIST) attempted to put some structure around this concept, by formally describing the Cloud in terms of five characteristics, three service models, and four deployment models, as shown in Figure 1 (NIST, 2009).
2.2 Characteristics

Based upon the definition in Section 2.1, it follows that Cloud Computing differs from traditional computing in a number of areas. In fact, NIST have defined five characteristics that are essential and particular to the Cloud model (Mell & Grance, 2011). These are briefly described as follows:

- **On-Demand Self-Service** – The Cloud consists of providers and consumers, and in such an environment, it should be possible for the consumer to request and consume resources whenever these are needed, without the need of explicit intervention by the provider. For instance, if an application running on the Cloud requires additional RAM or CPU time, then this application should be allowed to acquire these resources whenever needed, and for as long as is required. This allows the application to service requests easily, even when the application experiences temporary spikes in demand.

- **Broad Network Access** – In the Cloud model, most of the data is stored with the provider, and processing is also largely done on the Cloud. The consumer typically consists of a lightweight front end, which will allow access to these resources over a network. The services offered by the Cloud should, therefore, be accessible over a network, and from various types of hardware or software (Mell & Grance, 2011). Moreover, the service is typically “delivered using Internet identifiers, formats and protocols, such as URLs, HTTP, IP and representational state transfer Web-oriented architecture” (Gartner, 2009). A typical example is Gmail, which can be accessed over the Internet, from a wide variety of devices.

- **Resource Pooling** – Resources in the Cloud are shared and reused by several services, in order to enhance efficiency and build economies of scale (Gartner, 2009). By servicing several consumers using the same resources, these resources can be allocated as necessary in order to ensure that whenever possible, the various consumers’ demands are all met concurrently. The resources might potentially be spread across several physical locations, possibly even in different countries. The allocation and sharing of resources is typically abstracted away from the consumers, although they might be allowed to define some preferences such as in which country the data should be stored (Gartner, 2009). This might be done for performance reasons, or also to avoid legal and compliance complexities.

- **Rapid Elasticity** – Cloud Computing typically uses the concept that one pays for what he uses. This means that an application should be able to expand and contract its demand for resources as necessary and without warning. This concept is referred to as elasticity, and relates to any resource such as database usage, CPU time and security services, to mention some examples. In this respect, Cloud applications have been compared to a sponge – whilst a sponge will grow in size and soak up additional water, in the Cloud, the more the application’s demands grow, the more resources this will consume (Roggero, 2011). Elasticity, therefore, means that the model should “handle sudden, unanticipated, and extraordinary loads” (Fardone, 2012), and also contact its resources as necessary.

- **Measured Service** – Various payment models are common in the Cloud, such as pay-as-you go, fixed price plans or even free. Since the Cloud offers plenty of resources that the consumers can use as required, a way of metering and charging for the resources used should be possible. In fact, each type of pricing plan will typically be based upon the amount of resources consumed (for example the size of data written to disk or the number of database queries performed). Since “resource usage can be monitored, controlled, and reported” (Gartner, 2009), this then provides a fair and sustainable operating environment for the consumers and providers alike.
Apart from the five characteristics identified by NIST, CSA also identifies a further essential characteristic for Cloud Computing (Cloud Security Alliance, 2009):

- **Multi Tenancy** – In the Cloud, a single service instance can be used to serve several clients (also known as tenants). For instance, the Google Docs services are hosted on Google’s infrastructure, and these services are made available to several tenants. In the absence of multi tenancy, separate instances would be necessary for different tenants, resulting in duplication of software and/or hardware. When considering multi tenancy in the Cloud, it is important that although an instance on the Cloud caters for several tenants, data for different tenants should be isolated from each other and maintained securely. In fact, “multi tenancy in Cloud service models implies a need for policy-driven enforcement, segmentation, isolation, governance, service levels, and chargeback / billing models for different consumer constituencies” (Cloud Security Alliance, 2009).

### 2.3 Service Models

Since Cloud Computing is a vast model that incorporates many technologies and approaches, the term in itself is often subjected to abuse and misuse by vendors, who aim to benefit from the hype surrounding this technology. NIST has, therefore, attempted to clarify matters by breaking down Cloud Computing into the “Software, Platform or Infrastructure as a Service” (SPI) model (Mell & Grance, 2011). The SPI model, therefore, consists of three main approaches for Cloud Computing, with each one of these then possibly being further split into more specific types of implementations.

By analysing the main types of Cloud Computing service models, one notes that SaaS is mainly targeted towards end-users, PaaS is aimed towards software developers, while IaaS is of particular interest to enterprises (Monaco, 2012). New types of models are constantly being proposed in Cloud Computing, such as Storage as a Service (STaaS), Security as a Service (SECaaS), Data as a Service (DaaS), and API as a Service (APIaaS). These models, however, can all be considered to be sub classes of the main three types described by the SPI classification.

#### 2.3.1 Software as a Service (SaaS)

In this type of model, the user makes use of applications that are hosted and executed on the Cloud infrastructure. The Cloud infrastructure is basically “the collection of hardware and software that enables the five essential characteristics of Cloud Computing” (Mell & Grance, 2011) described in Section 2.2 of this study. In SaaS, a software application is deployed and managed by the Cloud provider, enabling the Cloud consumers to make use of this service without having to worry about the infrastructure needed to support it. The SaaS application is therefore hosted and made available to the Cloud consumers over a network, and it is centrally managed by the provider. There typically is no application to be installed at the client, since a thin client (such as browser) is used to execute the application over a network. A single instance of the software application is used by several consumers, usually with each user having access only to his data. SaaS means that users do not need to handle fixes and updates to the application, introducing a level of convenience. For example, in SaaS, hardware requirements, operating system compatibility, software updates and the rest of the Cloud infrastructure are all managed directly by the service provider. Examples of SaaS applications include Google Docs, Gmail and Microsoft Office365 which all run on the Cloud and provide the end user with some options to tweak the user experience without having to go into complex application settings and configurations.
Cloud Encryption and Key Management Considerations

SaaS is typically used for applications which are widely used by many consumers, but which usually only need to provide standard functionality that is common for all users (Kepes, 2011). Moreover, SaaS is mostly adequate for systems which, from a user’s point of view, are logical to be accessed through a web environment. Since these systems are hosted on the Cloud, they in turn also provide all the advantages that the Cloud’s elasticity and flexibility brings about with it.

Despite its clear advantages for the consumer, by moving its data to a SaaS system, the organisation might however get tied in with a specific vendor since it is then usually considerably difficult to change the provider and migrate existing data. Moreover, since the consumer would need to trust the Cloud provider to manage data storage, processing, security and the rest of the Cloud infrastructure, then the consumer must relinquish all of its control over the data once this moves to the SaaS application. Legal considerations also need to be taken into consideration when hosting the data externally.

2.3.2 Platform as a Service (PaaS)

The PaaS model provides a number of services that make Cloud software development possible, without the need for the organisation to worry about the underlying infrastructure that is required to develop, test and host its applications. Without PaaS, setting up and managing the infrastructure to provide services such as file hosting, databases and web hosting would be a complex and lengthy process. As with SaaS, this model provides access to various web-based interfaces for the organisation to consume and manage these services. Moreover, the service is shared with other Cloud tenants, introducing security concerns that need to be mitigated (Kepes, 2011). However, the fact that development, testing and hosting of the applications can be done on the Cloud infrastructure means that developers can leverage on the flexible and distributed nature of the Cloud, writing software which is both efficient and reliable. A typical example of PaaS is Microsoft’s Azure platform.

PaaS is particularly suitable for development environments which consist of large or distributed teams, possibly having dynamic structures which change. Aspects such as development environments, database services, hosting services and also testing and development services can all be efficiently provided by the Cloud using PaaS, which provides “the ability to automate processes, use pre-defined components and building blocks and deploy automatically to production” (Kepes, 2011). The application providers, however, often have some form of control over the configuration of the platform.

A limitation of PaaS is that the application would be tied to a particular provider, and its portability would be limited since it must exist on the Cloud. Moreover, some services might provide limited functionality or not support proprietary languages, which might be a limiting factor in some situations. Applications in PaaS must be based upon the technologies that are supported by the platform, something which can be a limiting factor at times. For instance, Microsoft’s Azure platform is now capable of not only running applications written in Microsoft dotNet languages, but also those based on Java since Microsoft aims to have Java as a “first class citizen on Windows Azure” (Microsoft, 2010). PaaS also does not permit the Cloud consumer to access the hardware and operating system upon which the services are hosted, something which might be desirable during software development in order to complete or optimise a particular task.
2.3.3 Infrastructure as a Service (IaaS)

In this type of model, the Cloud infrastructure is made available for the client’s use. This might include virtual servers, networks, storage space and database servers, to mention some examples. Since this infrastructure is on the Cloud, the consumer can also benefit from the scalability and elasticity that the Cloud offers, enabling the organisation to scale its infrastructure when and as needed. The Cloud infrastructure can then be used just as any traditional computing infrastructure would, also allowing the user to install operating systems or application software. However, in IaaS the consumer does not manage the actual physical infrastructure, but rather makes use of the resources requested (Mell & Grance, 2011). The consumer is responsible for aspects such as identity management, authentication, operating system management and configuration of the resources made available in IaaS (ENISA, 2009).

In IaaS, the Cloud infrastructure hardware is typically shared between the different Cloud tenants, with the provider ensuring that there is logical separation between the infrastructures of the various users (Kepes, 2011). Rackspace is a typical IaaS example, which enables the Cloud consumer to source virtual servers with ease, having the required processing and storage capacities. IaaS, therefore, is particularly suitable when the organisation is not willing to invest in the purchase of the necessary hardware infrastructure, or when it cannot do so due to constantly varying levels of demand. IaaS, however, results in slower performance when compared to non-Cloud scenarios where the hardware infrastructure is hosted on site at the organisation.

2.4 Deployment Models

In the Cloud, either of the three service models can then be combined with any of the four deployment models that are defined by NIST (Cloud Security Alliance, 2009). These deployment models vary in the type of customers that are allowed to use the Cloud infrastructure. These four deployment models are the Public, Private, Community and Hybrid Clouds (Mell & Grance, 2011).

- **Private Cloud** – As the name suggests, in this deployment model, the Cloud infrastructure is used exclusively by a single consumer (for instance, one organisation). This organisation might not necessarily manage the infrastructure itself, but it is the only Cloud user. This closed environment provides greater security assurance compared to more open models (Mell & Grance, 2011), but might not be as vast and flexible as the other models. Moreover, apart from security reasons, private Clouds are commonly used when the data or application needs to adhere to strict regulatory standards, such as SOX or HIPAA, since private Clouds allow the data to be managed in a better and more reliable way, making it easier to guarantee privacy, and also to avoid cross-jurisdictional legal complexities, when compared to a public Cloud (Metha, 2012).

- **Community Cloud** – This type of deployment model is similar to a Private Cloud, but instead of being used by a single consumer, it is made available to a closed group of organisations that have some form of shared interest or business relationship. Such a Cloud facilitates interaction between the organisations involved, and allows for economies of scale to be achieved.

- **Public Cloud** – This type of deployment model is not as restrictive as the previous two models, since the Cloud is made available to a much wider group of users. Whilst greater economies of scale can be achieved, such Clouds might be more complex to manage securely (Mell & Grance, 2011) since the infrastructure is shared with potentially malicious users. The Cloud consumers also have less control over how the Cloud is managed, such as where data is actually stored and processed.
- **Hybrid Cloud** – This deployment model involves linking together two or more Clouds, using some standardised approach “that enables data and application portability” (Mell & Grance, 2011). Since the linked Clouds can, in turn, also have different deployment models, one needs to take into consideration the various security aspects that come into play at the interaction points of the different Clouds. Each type of deployment model has its own particular set of risks that need to be managed in order to ensure that security is maintained during Cloud interaction. For instance, data processed by these Clouds should be clearly labelled and categorised according to its source and intended use, to ensure that it is always treated to the required level of security.

Apart from these main types of deployment models, new derivatives are also appearing, such as the Virtual Private Cloud which is based upon a Virtual Private Network (Cloud Security Alliance, 2009). Since these deployment models all have different levels of risk, unless otherwise specified, the Clouds considered in this study are assumed to be Public Clouds, since these have the highest risk levels out of the four deployment models defined by NIST.

### 2.5 Taxonomy of Cloud Computing Use Cases

Up to a few years ago the Cloud was simply a new buzzword in computing, but nowadays this has evolved into something that has many practical implementations and whose popularity seems to be on the rise. The various use cases of Cloud Computing can be categorised through the use of taxonomies that have been put forward by various researchers in recent years. Whilst these taxonomies often share similar high level classifications, the levels beneath these are somewhat diverse. This is due to the fact that the Cloud is vast and incorporates many technologies, making it possible to provide different taxonomies which are all relevant and correct. Figure 2 shows a Cloud taxonomy that is proposed in this study, whose roots derive from the taxonomy provided by OpenCrowd (Open Crowd, 2012). The one presented in this study, however, is simplified in certain areas and expanded upon in others, taking ideas from taxonomies by (Goodall, 2009), (Woloski, 2008) and (Laird, 2009). An example product for each classification is also given in this taxonomy.
In the proposed taxonomy, **Infrastructure as a Service** covers the provision of the Cloud Computing infrastructure needed to host applications. This broad classification can be further split into a number of sub classifications:

- **Public Cloud** – Platforms that are made available to the general public
- **Private Cloud** – Platforms that are only available to a particular organisation
- **Data and Compute Grids** – Platforms that are based upon grid computing

The **Platform as a Service** taxonomy classification is concerned with the provision of a platform suitable for running applications. This can, in turn, be split into various more specific use cases:

- **General Purpose** – Platforms for development of general purpose applications
- **Business Intelligence** – Platforms for business intelligence and big data analysis applications
• **Integration** – Tools for linking together different Clouds and applications
• **Development and Testing** – Platforms for software development and testing
• **Databases** – Scalable database systems

Yet another high level classification provided in the proposed taxonomy is **Software as a Service**, which covers the use cases that focus on the provision of software applications that run on the Cloud infrastructure. This classification can be split into various sub categories, such as:

• **Billing and Financial** – Applications that handle user subscriptions, bills, taxes and expenses
• **Sales and CRM** – Applications that manage sales, commissions and corporate systems
• **Productivity and Collaboration** – Applications that help complete everyday office operations (such as word processing), and collaboration across different teams
• **Human Resources** – Applications for managing staff resources
• **Backups** – Applications for taking and restoring backups of data
• **Content Management** – Applications for managing the content of a website
• **Social Network** – Tools for managing and customising social networks

A further classification in the provided taxonomy is **Cloud Management Applications**, which covers the provision of applications that are used to manage the Cloud. Such applications can be split broadly into two main categories:

• **Data** – Applications that enable scalable storage on the Cloud
• **Computation** – Applications that manage the Cloud’s elastic computation resources

**Cloud Management Services** is yet another high level classification for Cloud use cases, which covers the provision of services that are used to manage the Cloud. This can, in turn, be split into several sub classifications:

• **Appliances** – Software that provides hardware acceleration for particular Cloud operations
• **File storage** – Services that can store files used when building internal Clouds
• **Cloud Services** – Services used to manage the entire lifecycle of a Cloud application
• **Security** – Services that provide security, such as antivirus scanning and malware protection
• **Fabric Management** – Services used to manage high performance unified computing

Based upon this taxonomy of use cases, it is clear that there are many uses for Cloud Computing. The Cloud is being used by various organisations in both the private and public sector, and there is no indication that its popularity will subside soon (Szabo, Scholten, Horchner, & Skilton, 2012).

### 2.6 Conceptual Models

As shown in the taxonomy provided in Section 2.5, there are many different configuration choices that need to be done in Cloud, with the service and deployment models to be used, being two of the main decisions. Due to the different features, advantages and disadvantages that each approach offers, information security in the Cloud cannot be evaluated in a generic way. It is often not possible to draw conclusions that apply to all types of Cloud implementations. For this reason, in order to facilitate the evaluation of Cloud information security in the rest of this study, a number of conceptual models are being proposed which are essentially typical Cloud setups. Whilst there is virtually an unlimited number of different Cloud setups that may exist in
Cloud Encryption and Key Management Considerations

reality, these models cover some of the most common scenarios. For each of these models, the main actors, assets and data flows involved are highlighted, in order to serve as the foundation for further analysis throughout the rest of this study.

2.6.1 Model A – Public SaaS

Conceptual Model A consists of a SaaS service model, running on a public Cloud. Such an example would be Gmail. This model is, in fact, the most common version of SaaS both for personal and commercial use. In this model, the roles of the main actors involved are as follows:

- **End users** – These users own and use the data placed on the Cloud
- **Technical staff at consumer organisation** – This actor is not directly involved in the Cloud’s setup and maintenance and, therefore, has a limited role in this model. This staff is primarily responsible for liaising with the Cloud provider if support is required.
- **Cloud provider** – This actor is directly responsible for all aspects related to the setup, administration, running, availability and security of the Cloud infrastructure, its users, the application and any data that is processed or stored by the application.
- **Third parties** – Third parties might be involved in order to perform supporting functionality, such as encryption or key management on behalf of the Cloud consumer or provider.

In this model, technical assets such as software and hardware, are mainly under the responsibility of the Cloud provider. The provider also requires human assets to support the Cloud. The data, on the other hand, is owned by the Cloud consumer but it is administered, stored and processed by the Cloud provider. Therefore, when a SaaS application in a public Cloud is used, there is a clear dependency of the Cloud consumer on the provider.

In this model, the data is entered by the end users, and it flows out of the organisation and onto the internet, where it reaches the Cloud provider. As soon as the data leaves the consumer organisation’s network, it crosses a trust boundary since it would have left the direct control of the organisation. On its way, this data may be operated upon by third parties, such as an encryption proxy. Once the data reaches the provider, it flows internally through the Cloud, being stored and processed as necessary, in order to meet the end user’s requirements. When the user opts to retrieve the data, this flows in the opposite direction, from the Cloud back to the consumer organisation. All Cloud processing is transparent to the end user.

2.6.2 Model B – Hybrid SaaS

Conceptual Model B is similar to Model A, in that it consists of a SaaS service model. However, in this model, the data is no longer stored exclusively at the Cloud provider, but instead it is split between the provider and the Cloud consumer organisation. Sensitive data is stored at the consumer organisation, while any other data is stored directly at the Cloud provider. An example of this model would be SalesForce, which allows the consumer to store confidential data in-house (Tay, 2011).

In this conceptual model, the following actors are considered:

- **End users** – These users own and use the data placed on the Cloud
- **Technical staff at consumer organisation** – Compared to Model A, this actor has a more engaging role to play in this model, since not only is it responsible for liaising with the Cloud provider in the case of support, but it also performs administration and
Cloud Encryption and Key Management Considerations

maintenance of the private component of the Cloud. This includes the management of onsite data storage. This model assumes that the consumer organisation has a dedicated security team, which is a justifiable assumption seeing that hybrid Clouds are something which typically only large organisations would consider, due to their complexity.

- **Cloud provider** – As in Model A, this actor is directly responsible for all aspects of the public portion of the Cloud, including the setup, running and security of the Cloud infrastructure, the application and all data stored at the provider.

In this model, technical, data and hardware assets are therefore a shared responsibility between the Cloud provider and consumer. Human assets are also found on both sides, although the provider might require more human assets to handle larger amounts of clients and data. In this model, the data is still owned by the Cloud consumer, who is also responsible for the administration, storage and processing of sensitive data. The rest of the data is managed and stored by the Cloud provider. The dependency between the Cloud consumer and provider is stronger than in Model A, since in this model, the consumer provides the sensitive data, while the provider supplies the software application.

In this model, the data is entered by the end users, and it flows out of the organisation and onto the internet, where it reaches the Cloud provider. Here it can be processed by the software application on the Cloud, after which it gets stored. If it is sensitive data, this is sent to consumer where it will be stored, while any other data flows through the Cloud until it reaches the storage servers. The Cloud provider is less trusted than the consumer organisation itself and, therefore, as data is transmitted between these two entities it crosses a trust boundary. Data retrieval is essentially the opposite, where the data flows from the Cloud back to the consumer. The data flows in the hybrid Cloud are largely transparent to the end user, although the consumer’s technical staff need to have a clear understanding and a significant degree of involvement in setting up and managing these flows.

### 2.6.3 Model C – Public PaaS

Conceptual Model C consists of a PaaS service model, which runs on a public Cloud. An example of such a model would be Google Apps, which provides developers with various services than can be used to build their own applications that are based upon the Cloud. The main actors involved in this model are as follows:

- **End users** – These are the users who consume the applications that run on the PaaS Cloud, and who ultimately own and use the data placed on the Cloud.

- **Technical staff at consumer organisation** – This model assumes that the PaaS services are consumed by an application which is written by technical staff at the Cloud consumer organisation. These employees are not directly involved in the setup and maintenance of the services consumed, but they are responsible for creating a system which interacts with them. These users might not be security experts, but it is assumed that they have some form of security awareness in order to orchestrate together these Cloud services in a secure manner. This actor liaises with the Cloud provider if support is required.

- **Cloud provider** – This actor is directly responsible for all aspects related to the setup, running and security of the PaaS services offered. This model assumes that one of the services used is data storage, where the Cloud provider is entrusted with the responsibility of managing and storing the consumer’s data.

In this model, hardware assets as well as the PaaS services are therefore mainly under the responsibility of the Cloud provider. The software which interacts with these services is under the
responsibility of the Cloud consumer. Human assets are required by both the provider and the consumer, in order to develop and maintain a suitable system based upon the Cloud. The data is stored by the Cloud provider, but it is owned by the consumer organisation making use of the PaaS system.

It is assumed that in this conceptual model, the application which invokes the public PaaS services is running on a machine within the organisation’s perimeter. Data is entered into this application by the end users, and this then flows out of the organisation and onto the internet, crossing the trust boundary until it reaches the Cloud provider’s PaaS services. These services might be responsible for modifying, processing or storing the data on the Cloud. Any data that needs to be retrieved by the user then flows back to the consumer organisation, in the opposite direction.

### 2.6.4 Model D – Private IaaS

Conceptual Model D consists of an IaaS service model running on a private Cloud. It is assumed that not only is the Cloud infrastructure dedicated to a single organisation, but that it is also hosted on premise at the Cloud consumer organisation. A typical example of such a setup would be a server virtualisation system, allowing the organisation to provide virtual machine instances to its employees. In this conceptual model, therefore, the Cloud consumer and provider are essentially the same entity. The actors involved are the following:

- **End users** – These users consume the IaaS services, and they own and use the data placed on the Cloud
- **Technical staff at consumer organisation** – This actor has an active role, since it is directly responsible for all aspects of the private Cloud, including its setup, administration and security of the Cloud infrastructure, as well as the secure storage of Cloud data. This model assumes that the consumer organisation has a dedicated security team which is responsible for ensuring security of the internal Cloud.

In this model, the Cloud is ring fenced within the organisation. Therefore, technical, data, hardware and software assets are under the responsibility of the Cloud consumer. Human assets and expertise are also required by the organisation, in its role as the Cloud provider. In this model, the data assets are owned by the organisation, who is also responsible for their administration, storage and processing. This means that data never needs to cross a trust boundary.

Being a private deployment model, the data entered by the end users flows through the organisation's internal Cloud network, where it is processed and stored. It never leaves the organisation, although it can be accessed by company insiders, such as technical staff who administer and support the Cloud. Data retrieval involves locating the data, such as a disk volume, on the internal network and having it processed by the IaaS platform. The results are then returned back to the end user through the internal network.

### 2.6.5 Model E – Public IaaS

Similar to Model D, conceptual Model E consists of an IaaS service model, but which this time runs on a public Cloud infrastructure. An example of such a model would be Rackspace which offers virtual machines through the internet, over an IaaS Cloud model. The main actors involved in this model are as follows:
Cloud Encryption and Key Management Considerations

- **End users** – These users make use of the IaaS services, and own the data placed on the Cloud.
- **Technical staff at consumer organisation** – This actor needs to consume the IaaS services in order to setup an environment, such as a virtual machine, that the end users can use. It is also responsible for interacting with the Cloud provider to source the required Cloud services. This model assumes that there is no dedicated security team at the Cloud consumer organisation.
- **Cloud provider** – This actor is directly responsible for all aspects related to the underlying Cloud infrastructure required to support the IaaS services, including its administration and security. Data volumes are under the responsibility of the Cloud provider.

In this model, hardware assets as well as the IaaS services are therefore under the responsibility of the Cloud provider. The management of the actual virtual machine instances running on the IaaS platform are, however, under the Cloud consumer’s responsibility. Human assets are required by both the provider and the consumer, in order to administer the underlying infrastructure and the use of the IaaS services, respectively. The data volumes are securely stored by the Cloud provider.

Data is added to the Cloud by the end users, as this crosses the trust boundary as it moves from the organisation’s internal network onto the internet, in order to reach the Cloud provider’s IaaS services. Within the Cloud, the data might be transmitted along different points of the Cloud infrastructure, for storage, processing and analysis. Any data that needs to be retrieved by the user flows back in the opposite direction.

### 2.6.6 Summary

The following table summarises the main properties and assumptions done for each of the five conceptual models that are being considered:

<table>
<thead>
<tr>
<th></th>
<th>Model A</th>
<th>Model B</th>
<th>Model C</th>
<th>Model D</th>
<th>Model E</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Model</td>
<td>SaaS</td>
<td>SaaS</td>
<td>PaaS</td>
<td>IaaS</td>
<td>IaaS</td>
</tr>
<tr>
<td>Deployment Model</td>
<td>Public</td>
<td>Hybrid</td>
<td>Public</td>
<td>Private</td>
<td>Public</td>
</tr>
<tr>
<td>Third Party Involvement</td>
<td>Optional</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Dedicated Security Team</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Infrastructure Managed By</td>
<td>Provider</td>
<td>Consumer &amp; Provider</td>
<td>Provider</td>
<td>Consumer</td>
<td>Provider</td>
</tr>
</tbody>
</table>

### 2.7 Benefits and Concerns of Cloud Computing

The concept of Cloud Computing differs significantly from the traditional computing environment, and this brings a number of benefits as well as new challenges that are particular to the Cloud. In the Cloud model, the infrastructure, application, data storage and processing is beyond the consumer’s direct control as this is managed by the provider. Although the consumer might be given some options as to how to configure the Cloud offering, he will not typically be able to control all aspects. For instance, in a public Cloud, the consumer does not know where exactly
the data is being processed and stored. This layer of abstraction from the consumer provides both a number of benefits as well as new challenges or concerns.

A recent study carried out by KPMG found “the main drivers of Cloud Computing to be cost savings, improved flexibility and better scalability” (KPMG Netherlands, 2010). The Cloud brings about a reduction in costs, since a consumer only pays for the resources requested and/or used. Moreover, capital expenditure and operating costs are greatly reduced or avoided when using the Cloud, compared to traditional computing environments. The financial benefits here are clear, since the organisation only pays for the resources it needs, and only when it needs them. The KPMG study also lists flexibility and scalability as main benefits of Cloud Computing, both of which are a result of the Cloud’s features of on demand self-service, rapid elasticity, resource pooling and broad network access. This same survey also found that the Cloud can help the organisation reduce the complexity of its IT infrastructure by passing this responsibility onto the provider, allowing the organisation to focus more on its business operations. Apart from these benefits, the Cloud also provides other advantages (Catteddu, 2010) (CloudWays, 2011) (Juno, 2009). A Cloud managed by a reliable provider will help achieve high levels of performance and availability, and this will also be typically backed by service level agreements and contracts between the parties involved. Disaster recovery plans will also be in place by the service provider to ensure continued operation in the case of a failure or any other untoward event. Cloud applications also allow for device independence, since a thin client such as a browser is used client side to access these services, from virtually any device that has access to the network.

However, apart from the benefits associated with the Cloud, one also needs to consider the challenges or disadvantages that this brings (Catteddu, 2010) (KPMG Netherlands, 2010). Privacy in the Cloud is a main cause of concern, since the data and infrastructure is not under the direct control of the consumer. As a result, the consumer needs to have some assurance that the data is kept secure and that privacy is guaranteed. Compliance and legal considerations are also challenges associated with the Cloud, due to its distributed nature. When the data and its processing are moved away from the data owner’s direct control and placed onto the Cloud, compliance (for instance, with the Data Protection Act) becomes more difficult to achieve, especially when the data crosses over into other jurisdictions. Moreover, since in the Cloud there is a heavy dependence upon the service provider, the consumer must trust this provider to have contingency procedures (such as disaster recovery) and quality standards in place. This dependence upon the provider also results in the possible danger of loss of business should the provider, for any reason, decide to stop operating. Such cases might occur because of technical, business or financial reasons (Shimba, 2010). Another concern when using Cloud Computing, is that migration to another provider might be difficult or even impossible. Unstable costs which fluctuate based upon actual usage of the Cloud, might also be considered to be a disadvantage in certain cases, as is the fact that in order to access the Cloud, one requires a reliable network connection that is of adequate speed.

By observing the list of concerns associated with Cloud Computing, one will realise that most of these revolve around security considerations. In fact, security is often considered to be the main cause of concern when an organisation chooses to move to the Cloud. This is because “matching internal security requirements with the Cloud Computing vendor’s measures and controls, proves to be difficult in practice due to discrepancies, lack of insight and insufficient expertise” (KPMG Netherlands, 2010).

Various researchers have attempted to categorise the security concerns of Cloud Computing (Gartner, 2008) (Cloud Security Alliance, 2009) (Gregg, 2010) (Shimba, 2010). Inspired by the work of these authors, a number of security risks or concerns are identified in this section, which can be broadly classified under data, management and provider related risks.
Data is of central importance in the Cloud and, therefore, this should be maintained as secure as possible. The Cloud is subject to a number of data-related security risks, which include:

- Loss of governance and lack of clarity of where exactly in the Cloud the data is physically stored, resulting in less control over the data
- Uncertainty over who has access to the data, as the data on the Cloud might, amongst others, also be available to insiders at the provider
- Poor data segregation from other users, and not maintaining the data as secure as possible through encryption and data classification
- Partial data deletion, which possibly might result in data being accessible by third parties after supposedly having been deleted
- Interception of data being transferred, for instance through man in the middle attacks and sniffing

Management-related risks also need to be taken into consideration when understanding Cloud security. Such risks include:

- Compliance with regulatory requirements and standards (such as COBIT, ITIL and the Safe Harbour Agreement), which vary in different countries
- Unclear approaches to auditing of data and operations
- Lack of sufficient employee training, which is particularly important seeing that the human factor can be one of the main issues in information security (Egan, 2005)
- How business continuity can be guaranteed in the case of an untoward event
- Difficulty to investigate inappropriate activity on the Cloud, since “Cloud services are especially difficult to investigate, because logging and data for multiple customers may be co-located and may also be spread across an ever-changing set of hosts and data centres” (Gartner, 2008).

The Cloud provider also introduces various security risks into the picture. Amongst others, provider-related security risks that one should consider when reviewing Cloud security include:

- Insufficient guarantees given by the provider vis-à-vis the level of service provided
- The long term viability and stability of the provider
- How security issues will be managed once an attack takes place
- Lack of liability accepted by the provider, as is the case of Google Apps where the terms and conditions clearly state that “Google has no responsibility or liability for deletion or failure to store any content and other communications maintained or transmitted through use of the service” (Google, 2012).

The security concerns of Cloud Computing, therefore, revolve around the fact that the provider is trusted to manage the infrastructure and data. By implementing security controls, the aim is to create a model whose level of security is at least similar to what the consumer would have had, should it have instead opted for the traditional computing approach (Gregg, 2010).

### 2.8 Conclusion

The Cloud supports a vast number of different configurations, the choice of which depends upon the required area of application. Amongst the decisions that need to be done, one needs to choose from the various service and deployment models that exist. Irrespective of the choices done, however, the Cloud should always have the essential characteristics of on-demand self-service, broad network access, resource pooling, rapid elasticity, measured service and multi
tenancy. These characteristics bring about with them various advantages and disadvantages, something which was introduced in this Chapter. This Chapter also discussed these characteristics, as well as the different service and deployment models. A taxonomy for Cloud use cases was also proposed, in an attempt to structure the various types of Cloud implementations into a more manageable way. Moreover, a set of conceptual models which consist of typical Cloud configurations was also described, in order to guide further discussion in this study about Cloud information security.
3 Cloud Security

As discussed in Chapter 2, security is a major issue that can determine the success or failure of the Cloud. For this reason, this chapter will delve deeper into Cloud security, and highlight what role encryption and key management can play in achieving this security.

Although Cloud Computing is a new approach, the concepts behind it are not and the controls used to address security issues are often similar to those used in traditional computing environments (Cloud Security Alliance, 2009). However, due to the way that the Cloud is structured (such as the various service and deployment models), the Cloud introduces new risks into the picture. Organisations that make use of the Cloud must aim to identify these risks and ensure that they are addressed through the necessary security controls, be they technical, physical, procedural or managerial. In such a way, the concepts of confidentiality, integrity and availability can be achieved in the Cloud. How the provider and the consumer split this responsibility depends upon a number of factors, including the type of service and deployment models being used. For instance, in SaaS, the provider is typically responsible for managing security of the whole application and the underlying infrastructure, and such terms are agreed between the provider and the consumer. On the other hand, in IaaS, the provider is responsible for less of the infrastructure, so whilst he is in charge of the underlying setup, the consumer is responsible for ensuring higher level security. PaaS is a balance between SaaS and IaaS, where platform security is managed by the provider, but application security is managed by the consumer (Cloud Security Alliance, 2009).

Section 3.1 of this study mentions the key organisations that issue standards and recommendations in the area of Cloud Computing security. An analysis of the major security threats that the Cloud is exposed to is then carried out in Section 3.2. The perceived level of importance of these threats to organisations is then analysed further in Section 3.3, by reviewing and interpreting statistical research that has been carried out in the area of Cloud security.

Organisations should also carry out risk assessments, in order to mitigate Cloud security risks. In Section 3.4, an approach to guide such risk assessments in the Cloud is proposed. This is referred to as the Cloud Computing Risk Assessment Method (CC-RAM).

3.1 Organisations Working on Cloud Security

Over the years, a number of organisations have taken up the responsibility for promoting security in Cloud Computing environments, some of which are more prominent and authoritative than others. The Cloud Security Alliance (CSA) and the European Network and Information Security Agency (ENISA) are among the forerunners.

The CSA is a non-profit organisation which was setup “to promote the use of best practices for providing security assurance within Cloud Computing, and provide education on the uses of Cloud Computing to help secure all other forms of computing” (Cloud Security Alliance, 2012). This organisation is made up of various stakeholders, such as security experts, organisations and other associations. As the Cloud is constantly advancing, CSA continues to perform research and issues guidance on new and current Cloud security aspects. In (Cloud Security Alliance, 2009), CSA identifies a number of areas which need to be considered carefully in order to provide this security. Known as the thirteen security domains, these areas can be split into three main categories.


- **Cloud Architecture**
  - **Cloud Computing Architectural Framework** – Provides a structure around the Cloud Computing model, and this is based upon the Cloud model defined by NIST

- **Governing in the Cloud**
  - **Governance and Enterprise Risk Management** – Deals with how an organisation manages risks that are posed by the Cloud
  - **Legal and Electronic Discovery** – Discusses the legal issues involved in Cloud Computing
  - **Compliance and Audit** – Identifies how compliance and auditing should be handled when dealing with the Cloud
  - **Information Lifecycle Management** – Discusses the lifecycle of information processed by the Cloud, ranging from its creation and use, up to its termination or loss
  - **Portability and Interoperability** – Focuses on the aspects involved when a customer decides to change the Cloud provider, and what must be taken into considerations

- **Operating in the Cloud**
  - **Traditional Security, Business Continuity and Disaster Recovery** – Provides information about how an organisation can manage the risks in Cloud Computing to ensure continued business operations in the case of an unwanted event occurring
  - **Data Centre Operations** – Provides information on how to evaluate the Cloud’s data centre processes and its reliability
  - **Incident Response, Notification and Remediation** – Discusses the difficulties involved in handling incidents that take place in the Cloud
  - **Application Security** – Focuses on how security can be achieved in Cloud applications, across the different types of platform models
  - **Encryption and Key Management** – Provides information about why encryption and key management should be used to protect data and other resources in the Cloud
  - **Identity and Access Management** – Discusses the issues that an organisation might encounter when implementing identity management in the Cloud
  - **Virtualisation** – Provides information about the risks of virtual machine environments and other forms of multi tenancy

Another organisation that focuses on information security is ENISA, which works for the European Union institutions and its member states. Unlike CSA, ENISA does not focus only on Cloud Computing, but rather it deals with a variety of topics with the aim to set a common direction for security within the European Union (ENISA, 2012). In (ENISA, 2009), this organisation discusses a number of requirements that are needed for information assurance in the Cloud. Personnel security, supply chain assurance, operational security, identity management, asset management, data portability, business continuity management, physical security, environmental controls and legal requirements are all discussed within this report.

Apart from CSA and ENISA, other organisations regularly publish guidelines or research about Cloud Computing. Amongst these are Gartner, NIST, the Information Systems Audit and Control Association (ISACA), as well as the Big Four Auditors - KMPG, Deloitte, PWC, and Ernst and Young. With such a wide variety of big organisations speaking about Cloud Computing, one can clearly see the importance that this plays in today’s technological environment, and in the foreseeable future.
3.2 Cloud Security Threats

Information security is never easy to achieve, and this is especially true in a model such as Cloud Computing, which poses a number of additional inherent challenges as discussed in Section 2.5. In order to manage this security in the Cloud, risk assessments should be used to be able to then make informed decisions (refer to Section 3.4). However, to be able to perform such a risk assessment, it is necessary to first identify the threats associated with Cloud Computing.

Information security threats typically can be classified as being of one of three main types. These include threats to confidentiality integrity and availability. Threats to confidentiality include those threats that effect how confidential data is stored and managed in the Cloud. A threat to confidentiality means that data may be disclosed to a third party that is not authorised to access the data. Threats to integrity, on the other hand, mean that the integrity of the data might be compromised since it could be tampered with by users who are not authorised to modify the data. This threat covers the possibility of having the data modified or deleted, in part or in full, in the process compromising its security. A third generic classification of threats covers those which effect availability of the Cloud infrastructure, services or data stored within it. Problems with the availability of the Cloud or data, even if temporary, are a serious information security threat since these render the system or data unusable or inaccessible. Threats to availability might occur as a result of malicious or accidental events, or also due to insufficient resources or planning. Besides threats to confidentiality, integrity and availability, one can also mention the threat to compliance. This category of threats covers cases where for particular technical, procedural or managerial reasons, it becomes impossible to comply with legislation, policies or standards that the organisation should be following. Non-compliance can result in damage to the organisation’s reputation, as well as overall poor levels of information security within the organisation, which may also give rise to other security threats.

Organisations such as the Cloud Security Alliance and ENISA constantly analyse the security threats that Cloud Computing is subject to. The Cloud suffers from a number of threats which are unique to this computing model, and which essentially consist of specialisations of the generic information security threats (Cloud Security Alliance, 2010). The level of susceptibility of the Cloud to these threats is influenced by several factors, such as the particular Cloud models and configuration used. Aspects such as the service and deployment models, as well as the actors and data flows involved in the Cloud, have a strong influence on these threats. The main Cloud-specific threats are introduced in this Section, and these are then analysed using the conceptual models described in Section 2.6 to understand the impact of these threats in different Cloud setups. The threats introduced in this Section will then be analysed in more detail and expanded upon in Chapter 4 and Chapter 5 of this study.

3.2.1 Virtually Unlimited Compute Power

The Cloud infrastructure can be used for malicious purposes, by having an attacker exploit the virtually unlimited compute power and resources in order to launch a sophisticated attack or perform other malicious activities. This type of threat is particularly relevant in IaaS and PaaS models, where the user is allowed more control over the Cloud infrastructure, as opposed to SaaS where the user can only consume the Cloud resources through a hosted application that introduces a more controlled environment.

When considering the conceptual models previously proposed, Model A and Model B, which both consist of a SaaS service model, are therefore less vulnerable to this threat when compared to the other models. Out of these two, however, Model A is the most vulnerable to this threat since it
exclusive makes use of the vast resources that the Cloud provider has. Model C, which is a public PaaS model, suffers from this threat since the developer writing the system that consumes the various PaaS services can effectively orchestrate these services to exploit the virtually unlimited resources to launch attacks. Similarly, Model E, which is a public IaaS system, suffers from this issue since the end users effectively have close interaction with the Cloud’s infrastructure. This threat, however, does not really apply to Model D, which is an IaaS system built upon a private deployment model, since this conceptual model assumes that the Cloud infrastructure is hosted at the consumer organisation. As a result, in this model only the resources available on premise at the customer can be used to launch an attack. It, therefore, follows that this threat is particularly applicable to public deployment models, and to service models which allow close user interaction with the Cloud infrastructure.

3.2.2 Cloud Management

The Cloud itself needs to be accessed and managed, and this is typically done through a set of services and APIs which are exposed over the network. It follows, therefore, that if the security of these interface points is compromised, then the security of the Cloud in general will also be affected. Threats to confidentiality, integrity, availability and authentication of these management interfaces need to be considered. Conceptual Model D is the least vulnerable since it assumes a private infrastructure hosted at the consumer. However, in this model, the risk still exists since even internal employees might compromise the Cloud by attacking or exploiting the Cloud management services. Moreover, if the private Cloud is connected to a network that is exposed over the internet, an external user might also be able to attack the private Cloud by reaching this from a remote location.

Apart from threats to the Cloud APIs, threats to the processes of provisioning and de-provisioning Cloud user accounts also need to be considered. This applies to all conceptual models. In Model A and Model B, which assume a SaaS approach, the termination of a user account will probably also result in the purging of all data associated with this account. This is especially true for Model A, which is a public one, as opposed to Model B which is a hybrid system where sensitive data is stored at the client. A similar issue occurs in Model C and Model E, which are a public PaaS and IaaS model respectively, where the provisioning or de-provisioning of user accounts is also a likely target for attack. By provisioning several accounts, an attacker effectively gains more resources that can be used for malicious purposes. On the other hand, de-provisioning of a valid user’s account will result in unavailability of the system and loss of data for the legitimate Cloud consumer.

3.2.3 Cloud Provider Insiders

Malicious insiders at the provider are yet another threat to the data managed by the Cloud. These might have access to the data for legitimate purposes (such as management of the Cloud infrastructure), but these rights might also be used maliciously. The provider is trusted to take all the necessary precautions to avoid such attacks from occurring, for instance by screening employees and auditing all of their actions.

Conceptual Model A assumes a public SaaS system, where the provider is in complete control of the Cloud infrastructure and application. The threat of malicious insiders is, therefore, highly relevant in this model. In this SaaS implementation, the insiders can easily get to know where all their customers’ data is stored, and the structure in which this is saved. This standardisation makes it easier for the provider to maliciously access and understand its customers’ data. Model B, which is a hybrid SaaS implementation, suffers from the same problem as Model A when
considering the public component of the Cloud. Moreover, the private component is also vulnerable to this threat, although the insiders are now employees of the consumer organisation rather than of the Cloud provider. Whilst this threat also needs to be managed, the fact that the insiders are employees of the consumer organisation itself means that the organisation has more control over this threat. The same concept applies to Model D, which is a private IaaS approach.

Model C and Model E, which are public PaaS and IaaS systems respectively, are both vulnerable to the threat of malicious Cloud provider insiders. However, compared to SaaS, it is potentially more complicated for an insider to locate and understand the sensitive data in these models, since the consumer has more control over decisions of how the data should be managed. Such decisions include the location, structure and data encoding to be used.

### 3.2.4 Shared Cloud Infrastructure

The Cloud infrastructure is often shared between different consumers, and separation between these tenants is managed by the Cloud application or by the hypervisor. Although this layer between the user and the actual infrastructure aims to isolate the activities of the different users on the Cloud, the underlying hardware (such as the CPU) is ultimately still shared. This results in a possible threat whereby a guest operating system might exploit a vulnerability to gain malicious control over the Cloud infrastructure (Catteddu, 2010).

Conceptual Model E is the most vulnerable one to this threat. This is because it not only assumes a public infrastructure that is shared across the various Cloud tenants, but it is also based upon an IaaS service model that allows the users more control and direct interaction with the Cloud infrastructure. The threat applies in a similar way to Model C, which assumes a public PaaS system. However, being a PaaS service model, the user has less direct interaction with the Cloud infrastructure, making it less likely for a successful attack to be performed. However, despite this, the Cloud infrastructure is still shared in PaaS meaning that an attacker can still potentially exploit the shared infrastructure, although with less ease than in IaaS. Model D, being a private IaaS system, still suffers from this threat since the users are allowed closer interaction with the Cloud infrastructure. However, since the actors involved in this conceptual model are all insiders of the Cloud consumer, this threat is potentially less relevant to the organisation than in Model E.

Model A and Model B, which are a public and a hybrid SaaS system respectively, are also affected by this threat. However, since direct interaction with the Cloud infrastructure does not take place in these models, the chances of a malicious user exploiting the shared underlying hardware to gain control over the Cloud infrastructure, is less likely to occur and hence less of a concern.

### 3.2.5 Data Security

Data on the Cloud is subject to significantly more risks than in traditional systems which are more isolated and under the user’s direct control. Threats to data on the Cloud include deletion, modification and loss of confidentiality. Threats to any encryption used, also need to be considered. Amongst others, these include the use of weak keys, the lack of appropriate encryption algorithms and processes, weak key management practices and the loss of any encryption keys used. These issues are discussed in further detail in Chapter 5 of this study.

Conceptual Model D is the least vulnerable, since it is a private Cloud which is hosted at the consumer organisation. This model also excludes any third party involvement, meaning that any encryption services would be designed and controlled by the organisation itself. Whilst this does not exclude the possibility of having weak encryption or key management services implemented,
the fact that these services are under the direct control of the organisation and that the Cloud is isolated from external entities, means that such a threat has lesser impact on the organisation in a private Cloud. Data confidentiality, integrity or availability breaches might still occur in this model, but these would again be localised rather than being exposed over a wide public Cloud. The same can be applied to the private component of the hybrid Cloud considered in Model B, where sensitive data is stored only on premise. However, Model B also assumes interaction with a public Cloud, and this information flow across different actors in the hybrid Cloud can compromise the data’s security since it is exposed and more vulnerable than in a purely private Cloud.

Model A, Model C and Model E are all public Clouds with different service models, making them vulnerable to this threat since the data is placed on a public and shared infrastructure which is an attractive target for an attacker. Model A, which considers a SaaS model, is also potentially vulnerable to this threat since it is common to allow the provider to manage the data and encryption services in such applications. Whilst reputable providers will attempt to implement any encryption services in a secure manner, it can also be the case that the provider might not be sufficiently knowledgeable to do so especially if this is not within its field of expertise. Model C and Model E, which are a PaaS and IaaS system respectively, will typically implement more standardised approaches to encryption, such as the database encryption service provided by Microsoft SQL Server in PaaS and drive encryption provided by Microsoft Windows Server in IaaS. Such encryption mechanisms are well established, making them less likely to be poorly implemented. Key management, however, remains a possibly weak link in such approaches, meaning that data security might still be compromised in such conceptual models.

3.2.6 Access to Cloud Data

If the credentials to the Cloud are obtained by an attacker, the attacker may then “eavesdrop on [the user’s] activities and transactions, manipulate data, return falsified information, and redirect clients to illegitimate sites” (Cloud Security Alliance, 2010). This is a major threat, since the attacker might also use the information obtained from the Cloud in order to launch further attacks. Weak authentication, access control and authorization controls, therefore, pose a significant threat to the Cloud.

Conceptual Model D is the most resistant against this threat, since the Cloud is a private one. Eavesdropping might still occur by malicious internal employees or by external users who manage to gain access to the internal network over the internet or through some other network entry point. In contrast, Model E, which is the public counterpart of Model D, is more vulnerable to this threat. Authentication and authorisation controls implemented in Model E are managed by the Cloud provider and, therefore, accessible over the internet by several users, including malicious ones.

Model A, which is a public SaaS model, is vulnerable to this threat since such applications often make use of weak authentication methods such as username and password authentication, making it is relatively easy for an attacker to collect and use the user’s credentials. Apart from eavesdropping of unprotected network traffic containing these login credentials, techniques such as phishing can also be used to present the user with a malicious application that looks similar to the legitimate SaaS application. The threat also applies to Model B, which is a hybrid SaaS, since this Cloud also contains a public aspect. However, this threat becomes more complex in this model since there is an interaction between the public and private portions of the Cloud, during which authentication and authorisation are required to ensure data security. These controls
might also be misconfigured, such as a poorly setup firewall at the boundary of the private Cloud, allowing an attacker to circumvent the measures that aim to control access to the Cloud data.

Model C, which is a public PaaS system, is also vulnerable to this threat since it suffers from the disadvantages that a public Cloud presents. This model assumes that PaaS services are used by the Cloud consumer as the building blocks to create an application that runs of the Cloud. One of these services could be an online database service, which can be used to hold data that is specific to this custom-built application. Amongst this application data, one might also find user credentials and account details. For this reason, access to the PaaS services should be controlled to ensure no unauthorised access, since this might in turn result in other sensitive data being disclosed.

### 3.2.7 Unknown Risk Profile

Another threat that is faced on the Cloud is what is termed as an “unknown risk profile” (Cloud Security Alliance, 2010), meaning that when the provider is entrusted to manage the Cloud infrastructure, the consumer might not be exactly aware of how risks are being assessed and managed, and to what extent. The Cloud provider might also be making use of some non-standard technology or solution that is not sufficiently tried and tested, or it might have configured the Cloud infrastructure inappropriately, increasing further the exposure to risks. It can also be the case that the provider gives assurances to various stakeholders, which it might ultimately not be able to deliver especially if these are somewhat conflicting.

The unknown risk profile threat is particularly applicable to non-private deployment models. In fact, conceptual Model D is not affected by this risk, since the Cloud provider is essentially the consumer organisation and there is no third party involved. Any risk assessment and treatment plans would be under the complete control of the organisation, eliminating the chances of having an unknown risk profile.

Model A, which is a public SaaS system, is influenced by this threat since often the Cloud provider is entrusted with the data, the management of the Cloud infrastructure, and any other related services. The SaaS application might also be using non-standard technology or weak processes which the consumer might not be aware of. This conceptual model also allows for the use of third parties, such as proxies that perform services like encryption, but these might also introduce unknown risks in the process. Similarly, being a hybrid SaaS model, Model B is affected by this threat in the same way as Model A. The benefit here, however, is that in this conceptual model any sensitive data is stored at the Cloud consumer rather than at the provider, reducing the impact of this threat. Despite this, this conceptual model supports the flow of sensitive information from the private Cloud into the public portion of the Cloud, where it can be processed together with the other data. Unknown risks related to how the provider processes this information might also exist, and these need to be considered.

Model C and Model D, which are public PaaS and IaaS models respectively, are also vulnerable to this threat. However, since in these service models the Cloud consumer has more control over the infrastructure, the number of unknown risks might be less. Despite this, the Cloud provider still has an important role and a substantial degree of control, meaning that risks unknown to the Cloud consumer might still be present in these conceptual models.
3.2.8 Forensics

Auditing of actions done on the Cloud, as well as forensics, might be difficult or impossible to achieve at times, depending on how the Cloud infrastructure is setup. Therefore, when the consumer decides to move to the Cloud, the probability of not having the required forensic data available when necessary, must be considered (Catteddu, 2010).

Conceptual Model D, being a private Cloud completely under the organisation’s control, suffers the least from this threat. This is because the organisation can easily introduce auditing controls when and where required. It is, however, still possible for the organisation to misconfigure these controls resulting in missing forensic information in case of an incident. However, the likelihood of this occurring is similar to that of traditional, non-Cloud environments.

Since conceptual Model B is a hybrid system where sensitive data is stored on the private Cloud, the organisation can introduce auditing controls related to the storage of this confidential data. However, once the data leaves the private Cloud and moves into the public portion of the SaaS Cloud for it to be processed and used, then such forensic controls would be under the responsibility of the Cloud provider. Moreover, the public Cloud in this conceptual model is vaster and dispersed, with many information flows happening within the provider’s infrastructure. This makes the threat of not having the required forensic information when required, a reality.

Model A, Model C and Model E all suffer from this threat since they are all based upon public service models. However, the service model also plays an important role in this threat. Model E, being an IaaS service model, allows the user more direct interaction and control of the Cloud infrastructure, when compared to the other service models. This means that since only the lowest layers are entrusted to the provider, then the consumer can implement the required forensic controls at higher layers. Model A, on the other hand, is a SaaS model and, therefore, everything from the lowest layers of the Cloud infrastructure up to the application itself, is under the provider’s control. In this model, the consumer interacts with the Cloud application by entering the required information, after which it loses any control over the data and operations done on it. This, therefore, increases the probability of having Cloud processes or operations which are not audited, due to technical complexities or poor information security practices.

3.3 Perception of Cloud Security

Having understood the main threats that the Cloud is exposed to, it is useful to compare these threats to the concerns that security officers actually have when considering Cloud Computing. As Bruce Schneier, a renowned cryptographer and computer security specialist states, “security is both a feeling and a reality and they are not the same. You can be secure even though you don’t feel secure. And you can feel secure even though you’re not” (Shneier, 2008). Therefore, security is not only about how effective the actions taken to manage the risks are, but it is also concerned with how the users perceive security and the effectiveness of the measures taken.

3.3.1 Survey Methodology

In this Section, a number of surveys are referenced to understand better how organisations perceive Cloud security. The two main surveys used in this study are (KPMG Netherlands, 2010) and (Ponemon Institute, 2012).
KPMG Survey Method
In the Cloud Computing survey carried out by KPMG in 2010, over 100 Dutch organisations from various sectors of the industry were surveyed. The survey was completed by decision makers of the organisations in question. These organisations varied in size, with some employing less than 100 people, while almost half of the organisations surveyed employed more than 5,000 employees. The distribution is considered to be adequate for such a study, allowing for statistically significant results to be derived. The reader is referred to (KPMG Netherlands, 2010) for further details on the methodology used by KPMG in this survey.

Ponemon Institute Survey Method
In 2012, Ponemon Institute carried out a large-scale independent study across thousands of organisations in several countries. Over 4,000 organisations from the United States, United Kingdom, Germany, France, Australia, Japan and Brazil were surveyed in this study (Ponemon Institute, 2012). Only organisations that have “bona fide credentials in IT or security fields”, coming from a variety of industry sectors, were selected to ensure that the study is reliable. The organisations varied in size, ranging from small companies up to large companies employing several thousands of employees. The reader is referred to (Ponemon Institute, 2012) for a detailed review of the methodology used by Ponemon Institute.

Both of these surveys suffer from limitations which are inherent to all survey-based research. This includes (Ponemon Institute, 2012):

- Non-response Bias – Since not all organisations agree to participate in the survey, it might be possible that the organisations that did not complete the survey had a substantially different opinion than those that did respond
- Sampling-Frame Bias – Since a survey is not sent out to all organisations, it might be possible that the sample selected for the survey was not sufficiently representative of the rest of the population
- Self-Reported Results – The response provided by an organisation participating in the survey is not verified, and it is assumed that the organisations are truthful in their response. Anonymity and confidentiality is used in these surveys to encourage honest and accurate responses.

Although these limitations are inherent to all surveys done, the results obtained are nonetheless significantly relevant and representative. In fact, results obtained in both of these surveys seem to corroborate one another.

3.3.2 Survey Results

The Goldman Sachs Equity Research Report drawn up in 2011, which surveyed CIOs of several companies, found that 70% of the respondents were particularly concerned about the security of the Cloud. These perceptions were in line with the security issues identified in Section 3.2, since loss of control over the data and the key role of the third party provider were the most common concerns expressed (CipherCloud, 2011). This makes it clear that security is both a perceived and an actual issue when considering the Cloud.

Two studies, one by IDC (2009) and the other by KPMG (2010), corroborate this conclusion and also shed further light on the concerns that the participating organisations have about Cloud Computing. Figure 3 summarises the main findings of the KPMG study, showing that security issues (76%), legal considerations (51%), compliance (50%) and privacy issues (50%) ranked top on the list. These issues, therefore, need to be assessed and mitigated by the organisation, before
moving to the Cloud. The organisations, however, did not consider the fact that the Cloud is still a new technology as being a major concern (KPMG Netherlands, 2010). The same organisations also felt that they did not have sufficient in-house knowledge and expertise about the Cloud, with more than half quoting lack of security expertise as their main concern in this area. The fact that the Cloud is still a relatively new technology, therefore, also provides a set of unique challenges.

When the security issues were considered in more detail, the KPMG report found that two thirds of those surveyed considered these as sufficiently worrying to prevent the organisation from using the Cloud (KPMG Netherlands, 2010). The possible lack of security measures at the provider’s side was the main factor that contributed towards this percentage. This highlights the importance of the customer assessing the security aspects of various Cloud providers and then selecting the one that best matches its security requirements, before moving to the Cloud.

(Ponemon Institute, 2012) interestingly found that organisations which are security aware are typically more in favour of Cloud adoption, whilst organisations that are less security conscious fear the Cloud and its possible security issues. Figure 4 shows the security posture for organisations that store sensitive data on the Cloud, and that of organisations that do not. Here, the security posture of an organisation is defined using the Security Effectiveness Score (SES), which ranges between -2 and 2. SES is a technique developed by Ponemon Institute which is obtained by rating and averaging 24 security features or processes that the organisation can adopt, with encryption being one such example. A complete list of these security attributes can be found in Appendix A. A high SES indicates that the organisation takes security seriously, while a lower score indicates that in general the organisation is not security aware.

Although the scores vary slightly between different countries, the trend in Figure 4 is clear. Organisations that give adequate importance and attention to security aspects, consider the Cloud to be sufficiently secure for their business purposes after taking the security measures in line with their risk appetite. On the other hand, organisations which do not have a security culture in place, will not take the necessary security measures and perceive the Cloud as being too insecure for any actual business use.
Cloud Encryption and Key Management Considerations

Figure 4 - Relationship between Security Posture and Cloud Adoption
Source: (Ponemon Institute, 2012)

(Ponemon Institute, 2012) focusses on the protection of data, when delving deeper into the security issues of the Cloud. This study discovered that, despite the fact that security is a major consideration, half of the respondents still send sensitive or confidential data to the Cloud. A risk assessment using techniques such as the Cloud Computing Risk Assessment Method (CC-RAM) proposed in Section 3.4, should help the organisations clarify what risks are involved in doing so. The decision whether to move data to the Cloud or not depends not only on whether the organisation has a security culture embedded in its business operations, but also on how willing the organisation is to risk having a possible security issue affecting the confidential data in order to benefit from the convenience that the Cloud offers.

Moreover, it was discovered that 44% of respondents believe it is the provider’s responsibility to protect this data, compared to only 30% who place this responsibility on the consumer. However, despite the fact that the majority of the respondents state that the protection of data is the provider’s responsibility, 63% are not aware of what measures the Cloud provider is taking to do so. Unless these Cloud consumers are knowledgeable about the security measures taken by the provider, it is possible that the security measures actually taken are not aligned with those that the consumer expects from the provider. By understanding the Cloud provider’s approach to security, the consumer can also compensate for any gaps in the provider’s security, if necessary.

One of the factors which influences the organisations’ Cloud security perception, and which contributes to these survey results, is the fact that over the years numerous failures or security issues have been reported in the Cloud. Some of these cases have gained prominence and were reported extensively in the media, further raising concerns about Cloud security. One such example is the Microsoft Sidekick incident which occurred in 2009 and lasted for six days. Microsoft reported that a number of customers lost data from their smartphones, including emails, contacts and photos, due to a system failure (Cellan-Jones, 2009). Microsoft Azure also suffered a similar outage in 2009 (The Register, 2009). In 2010, SalesForce was disrupted leaving all of its users unable to use the system, although no data loss was experienced (Brooks, 2010). Similarly, in 2012 over 5.25 million Gmail users were affected after Google suffered a widespread outage which left many Google Apps and Gmail users without access to their data (Needleman, 2012).

Despite these current security concerns, however, it seems that good progress is being made in this area and that Cloud adoption will continue to accelerate. In fact, it is estimated that “by 2015, security will shift from being the number one inhibitor of Cloud, to one of the top enablers and drivers of Cloud services adoption” (Penn, 2010). Whilst security concerns are currently
limiting the adoption of Cloud Computing, once standardised approaches and frameworks to achieving security will be defined and refined, it is expected that the popularity of the Cloud will surge significantly. This is because the consumers would then be able to reap the benefits that the Cloud provides, knowing that this does not introduce security issues which are far greater than those of traditional computing environments. In fact, a survey held by IDC found that organisations are interested in pursuing the Cloud model for various areas of implementation, such as collaboration (67%), web application (67%), backups (60%), business applications and personal productivity, to mention some examples (IDC Enterprise Panel, 2009). The reader is referred to Section 2.5 of this study, which describes a comprehensive taxonomy of these Cloud adoption areas.

3.4 Cloud Computing Risk Assessment Method

In information security, a risk is defined as a “characterisation of the danger of a vulnerability or condition” (Swiderski & Snyder, 2004). Since the Cloud is subject to many threats and risks, it is important that these are evaluated in order to be able to make informed decisions that ensure security is achieved. This is true not only for the Cloud, but also for any area of information security, so much so that risk assessments are commonly used in these areas. A risk assessment involves a methodical evaluation of all security risks, such that the expenditure on controls can then be balanced against the possible harm caused by the risk. However, owing to the nature of the Cloud and to the different deployment and service models (refer to Chapter 2), the Cloud risk assessment process is more complex than that of traditional IT systems, as different Cloud architectures bring about different levels of risk (Catteddu, 2010).

The Cloud risk assessment process consists of two main parts – risk analysis and risk evaluation. The risk analysis process involves identifying the possible sources of risk, as well as the threats that could have negative impacts. This process also considers the likelihood of the event occurring and the level of impact that this would have. Risk evaluation then involves comparing the calculated risk levels to some risk criteria, in order to understand the significance of the risk. Various risk assessment frameworks and models have been put forward, which cover different parts of Cloud risk assessment (Cloud Security Alliance, 2009) (Catteddu, 2010) (International Standards Organisation, 2005). Based upon these models, a complete process for analysing the Cloud’s risks is put forward in this study. This proposed technique is called the Cloud Computing Risk Assessment Method (CC-RAM), and is depicted in Figure 5.
Asset Identification
The first phase of CC-RAM involves identifying the assets involved in the process that is being analysed. These assets are broadly categorised into two types, namely data and processes, since consumers are usually either transmitting information to or from the Cloud, or else performing some processing on the Cloud (Cloud Security Alliance, 2009). An organisation that is using the Cloud might will not necessarily have all of its assets migrated to the Cloud, but it will typically only shift some of these assets, keeping the rest under its direct control. Therefore, it is important to clearly understand which assets are involved in the Cloud and which are not, in the initial phases of the risk assessment process.

Asset Valuation
Once the assets involved in the Cloud are identified, these need to be evaluated in order to understand their level of importance to the organisation. This process can either be done at a high level, or else in more detail using a valuation methodology. During this phase of CC-RAM, the confidentiality, integrity and availability requirements of the asset being valued are considered, with the aim of understanding what impact any damage, loss or harm to the asset will have to the organisation (Cloud Security Alliance, 2009). Clearly, assets that are of significant importance to the organisation will need to be treated with more care throughout the rest of the risk assessment process.

Deployment Model Analysis
Once the assets are identified, CC-RAM requires that the Cloud deployment model is considered. As described in Section 2.4, different deployment models include private, public, community and hybrid models, with each one having different levels of risks. Whilst a private system might provide the least risks, it could also be the case that it is the least flexible or scalable model. During this phase, each type of deployment model should be considered vis-à-vis the asset’s value to organisation, such that the most appropriate Cloud deployment model can be selected for the assets in question (Cloud Security Alliance, 2009).
Cloud Encryption and Key Management Considerations

Service Model Analysis
CC-RAM also involves analysing the different Cloud service models. As discussed in Section 2.3, SaaS, PaaS and IaaS provide the consumer with different levels of control on the data, its processing, and the Cloud infrastructure. These models also provide different opportunities for risk mitigation (Cloud Security Alliance, 2009), some of which place more responsibility on the provider, while others place more responsibility on the consumer. At this point of the risk assessment, these service models should be considered, and the model which best suits the security requirements of the asset being considered, should be selected.

Data Flow Identification
As part of any information security risk assessment, it is essential to also understand the data flows involved, as these need to also be evaluated and secured. In the Cloud model, one needs to not only consider the data flows that are internal to the Cloud, but also data that flows in and out of the Cloud. By understanding these flows, it becomes easier to “identify risk exposure points” (Cloud Security Alliance, 2009) that need to be managed. Whilst it is usually easier to identify the need to protect data that is stored in the Cloud, the risks associated with the transmittal of data in and out of the Cloud might not be immediately evident, and data flow identification can help identify such risks.

Threat and Vulnerability Analysis
Once the assets have been identified, and the Cloud infrastructure to be used has been determined and understood, the next phase involves identifying the particular threats and vulnerabilities that affect these assets. The Cloud is subject to a number of threats, some of which are particular to the Cloud while others are more generic. The reader is referred to Section 2.5 and Section 3.2 for a more detailed analysis of these threats. The probability of a vulnerability being exploited is also determined, taking into consideration the technical complexity of attack, technical capabilities of attacker, likelihood of threat and susceptibility of vulnerability to be exploited (International Standards Organisation, 2004). Techniques such as STRIDE threat analysis and DREAD vulnerability analysis can also be used during this process (Microsoft Corporation, 2003).

Risk Score Calculation and Risk Treatment Planning
A risk score is then calculated based upon the previous findings, taking into consideration the impact, threat and probability to assigning ratings for the various threats and vulnerabilities. Risk matrices and tools such as the CCTA Risk Analysis and Management Method (CRAMM) (Yazar, 2002) can be used to facilitate this process, which assigns ratings between one and seven. The risk scores obtained are then used to decide upon a risk treatment plan. Risks that score high should be considered to be more urgent and critical than lower scoring risks. However, the risk treatment plan is also affected by the organisation’s risk appetite and this will influence the decision to mitigate the risk, accept it, transfer it, or else ignore it. Transferring of a risk in the Cloud might involve passing on the risk to the Cloud service provider.

Countermeasure Identification, Gap Analysis and Control Implementation
The countermeasures that are necessary to mitigate the risks are then identified, which should be based upon standards and best practices. Chapter 6 and Chapter 7 discuss various best practices and measures related to encryption and key management, which can be used to address identified risks. When considering the countermeasures to be implemented, one must weigh the cost of the countermeasure against the benefit that this provides. For instance, in CRAMM, each countermeasure has associated with it a risk ratings, making it easier to choose the appropriate controls to mitigate risks with different risk scores (Yazar, 2002). Once the necessary countermeasures are identified, a gap analysis is then performed to compare these controls with what is already implemented. In doing so, the organisation can then determine what
countermeasures need to be implemented, and it can also analyse its options on how to implement any missing ones. The list of required countermeasures is then prioritised, and these are then implemented by either the Cloud consumer or else by the provider. As part of the risk assessment, it might also be the case that the need is identified to change the Cloud model being used (refer to Section 2.3), to improve upon the Cloud infrastructure, or else to even change the Cloud provider used.

**Feedback Loops**

The CC-RAM approach allows for two feedback loops. After implementing the Cloud security controls, the process of identifying the threats, assessing the risks involved, defining the necessary countermeasures, and implementing them after performing a gap analysis, should be repeated regularly in order to ensure that any new or previously unhandled risks are identified and addressed as soon as possible. Such feedback loops should be a continuous process carried out by the organisation. Moreover, CC-RAM in itself is a cyclic model, meaning that once the whole risk assessment process is complete and the controls are implemented, the organisation should eventually start again from the first step of the model, identifying its assets and questioning the Cloud’s suitability and security. Although it is typically not necessary to restart the CC-RAM cycle too often, it is sometimes necessary to do so to ensure that the best approach to Cloud Computing is selected by the organisation. Policy defined by the organisation dictates how frequently each of these two feedback loops should be carried out.

When a risk assessment is being done by the Cloud consumer, the consumer might not have sufficient influence to force the provider to implement particular countermeasures. In such cases, the consumer might have to find alternative countermeasures that can be implemented client side, or else seek assurance from the provider through the likes of service level agreements and contracts.

**3.4.1 CC-RAM Design Decisions**

When designing the CC-RAM approach to risk assessment, various design decisions had to be done in order to ensure that such a technique is effective, usable and easily adopted by organisations that already have other risk assessment procedures in place. For this reason, CC-RAM is based upon established standards and techniques, which are packaged together into a single model.

The CC-RAM approach starts off based upon recommendations by the Cloud Security Alliance (Cloud Security Alliance, 2009). Although the Cloud Security Alliance does not define a full risk assessment methodology, it provides some guidelines on how to evaluate assets within a Cloud Computing environment. As discussed in Chapter 2, the Cloud provides several deployment options, which means that any Cloud risk assessment model should cover and consider all of these combinations. The recommendations put forward by CSA help understand the Cloud environment in question and, therefore, this approach was incorporated directly into CC-RAM.

In line with other risk assessment models, the first steps of the model involve identifying and valuating the assets that are being assessed. The decision to include this as the initial phase of CC-RAM is that, for this model to succeed, it is imperative that the organisation clearly understands what is being assessed and what importance this has to the organisation. Clearly, important assets will require more protection than assets which are of trivial importance. These phases of the model also allow the organisation to define the boundaries of the risk assessment process, ensuring that its assessment will be more focussed. If no risk assessment boundaries are established as part of CC-RAM, the scope of the analysis would risk being too wide, making it
difficult for the organisation to perform a detailed and complete assessment. When it comes to asset valuation, CC-RAM intentionally does not specify explicitly how this is to be accomplished. Whilst some organisations might opt to perform a high level and informal valuation exercise, other organisations might already have detailed and established valuation processes in place. Moreover, the organisation might have already valued the same asset for other purposes, using its established processes. Therefore, CC-RAM is flexible by allowing for any valuation methodology to be used, and it also allows for reuse of existing asset valuation information that it might have.

In line with the Cloud Security Alliance’s recommendations, CC-RAM then requires the user to evaluate the potential Cloud deployment and Cloud service models that can be used. This step is essential since it converts a risk assessment into one that is specific for the Cloud. By considering the different Cloud models, the organisation can then determine which approach to take after weighing the security issues against the benefits of each model, and comparing this with the asset’s value. These phases of CC-RAM aim to get the organisation to think about the suitability of the Cloud for the asset in question, and to understand what level of control the organisation will have on the asset’s security once it is placed on the Cloud. At this point, the model starts moving from a high level analysis to a lower level one, by considering the Cloud data flows in question. No specific methodology to identify these data flows is provided, since typically this necessitates that the organisation goes through the entire lifecycle of the asset and manually identifies the flows. Despite the fact that this sounds conceptually simple, the importance and difficulty of this phase should not be underestimated.

At this point, CC-RAM shifts from the Cloud Security Alliance’s recommendations, to the risk assessment processes defined in ISO13335 and ISO27001, which are more traditional and standardised methodologies. This is done intentionally, since organisations that already follow standards such as these can then easily integrate CC-RAM in their established risk assessment processes. Moreover, in itself, ISO27001 provides a number of inherent advantages which the organisation stands to benefit from. Amongst others, these include:

- It is an internationally recognised approach to security management
- It is a widely adopted best practice which is tried and tested, and created by experts
- It is sought after by customers and business partners
- It builds a level of confidence for the business to invest in security
- It helps the organisation review and maintain information security
- It helps meet legislation requirements
- It can lead to certification

By following the risk assessment methodology used in ISO27001, CC-RAM can help an organisation that is already following this standard to easily integrate Cloud risk assessment in their processes. On the other hand, organisations that do not already follow this standard stand to gain from some of the advantages that this offers, since through CC-RAM at least part of the standard’s processes will be followed.

During threat and vulnerability analysis, CC-RAM allows the organisation to use any approach that it considers to be suitable. Typically, organisations will already have some form of threat and vulnerability processes established, such as ISO13335, STRIDE and DREAD. Irrespective of which approach is taken, ultimately these all provide a similar outcome. It would, therefore, be unreasonable for CC-RAM to dictate which approach must be followed, as this would be an added burden on the organisation.
Cloud Encryption and Key Management Considerations

3.5 Conclusion

Despite its advantages, the Cloud is subject to various threats which need to be identified, analysed and understood in order to be addressed using appropriate measures. Apart from the generic information security threats of confidentiality, integrity and availability, one needs to also consider Cloud-specific threats. Amongst others, these include the threat of an attacker having virtually unlimited Cloud compute power, as well as the threat of attacks on Cloud management interfaces. Cloud provider insiders and the fact that the Cloud infrastructure is shared by different tenants, are also a cause of concern. Threats to the data’s security and to its access control mechanisms should also be considered during a Cloud threat assessment. The Cloud also provides the threat of having an unknown risk profile and the lack of sufficient forensic and auditing information. The applicability of these threats to the Cloud depends on many factors, including the service and deployment models used.

A risk assessment should be carried out to understand and analyse these threats and the risks that the Cloud introduces. The Cloud Computing Risk Assessment Method (CC-RAM) provides a structured way to perform a risk assessment that is adapted to the Cloud. This model starts off
with asset identification and valuation, after which the Cloud deployment and service models are analysed. The data flows involved as well as the applicable threats and vulnerabilities are then identified, in order to help rate the risks involved. A risk treatment plan is then drawn up, countermeasures are identified, and these are then implemented following a gap analysis. Feedback loops in CC-RAM help ensure that Cloud risks are regularly evaluated and addressed using suitable countermeasures, in order to reduce the degree of risks that the Cloud provides.
4 Encryption and Key Management in the Cloud

Encryption is defined as “the task of transforming data such that it is unintelligible to an outside observer. If used successfully, encryption can significantly reduce chances of outside interception and any possibility of data modification” (Dhillon, 2007). Cryptography has been a subject of interest for many centuries, since man has always wanted to prevent confidential information from being disclosed to unauthorised people. For instance, Herodotus, an ancient Greek historian, describes how the Greeks used to transmit hidden messages by tattooing the message on a messenger’s shaved head, and then allowing his hair to grow before sending him to his destination (Singh, 1999). The key to retrieving the message was to then shave the messenger’s head, exposing the tattoo.

Even though encryption has become much more complex and sophisticated since ancient times, the principle remains the same. During encryption, the data is obfuscated or encoded using a cryptographic algorithm in such a way that only the intended receiver, who owns the correct key, can reverse the process. Encryption thus provides confidentiality, by limiting access to the actual information content, preventing it from being disclosed to unauthorised users. The encryption algorithm relies on computational complexity, making it infeasible to break using unintelligent techniques.

Whilst encryption is not a panacea in information security, in some cases encryption and key management practices might be considered to be appropriate countermeasures to address particular risks. In the Cloud, both the consumer and the provider want to ensure that the data remains secure, and encryption can help address this. In fact, sometimes encryption is also required by law or it may be useful for achieving compliance, such as with the Data Protection Act (United Kingdom Act of Parliament, 1998). Whilst encryption in itself does not prevent data from being lost or stolen, the fact that this is encrypted makes it much more difficult for the data to be used by someone other than the intended recipient. If properly encrypted data were to be lost, legislation such as the Safe Harbour Act would still consider this data to be secure (European Union Data Protection Directive, 1998).

In this Chapter, the use of encryption to address particular security risks is described. Section 4.1 introduces the topic of cryptography and how this can be used in the Cloud. More specifically, encryption as a possible threat countermeasure is discussed in Section 4.2, which also refers back to the threats and CC-RAM approach for risk assessment that were introduced in Chapter 3. Section 4.3 discusses the fact that in order to mitigate the threats, encryption can be applied to data that is at rest, in transit or in use. The three main Cloud service models, namely SaaS, PaaS and IaaS are revisited in Section 4.4, which discusses the various approaches to applying encryption to each of these models. Finally, Section 4.5 concludes by analysing the trends in Cloud encryption, in order to provide a better understanding of what the current situation on Cloud encryption really is.

4.1 Cryptography Services Supporting the Cloud

As part of its work, CSA looks into how cryptography can be used in the Cloud in order to implement information security primitives, mechanisms and services (Security as a Service Working Group, 2011). Whilst a Cloud implementation might not necessarily make use of cryptography to implement all of these services, cryptography certainly plays an important role in achieving information security, when used in conjunction with other procedural and technological measures. The decision of whether to implement a security measure based upon cryptography depends on the findings of a risk assessment and the results of a gap analysis.
When carrying out a risk assessment, possibly using the CC-RAM process described in Section 3.4, the various threats and risks that the Cloud is subject to are identified. The exact nature of these threats depends upon the scope of the risk assessment and the particular scenario being investigated (refer to Section 3.2). When following CC-RAM, the assets to be assessed are first identified and valued, after which the Cloud deployment and service models are selected and analysed. These decisions, together with the data flows involved, affect the results obtained during threat and vulnerability analysis. One can then compare the Cloud threats identified during this analysis with the cryptography-based measures, in order to determine which of these threats can be addressed at least partially using cryptography. Whilst it is evident that cryptography on its own will not address all possible threats, it can be effectively used to address a number of security concerns, especially when used in combination with other countermeasures. In particular, cryptography can be used to provide confidentiality and integrity. These can be achieved through various services, such as user authentication, data origin authentication, data integrity, non-repudiation and confidentiality (Certicom Corp, 1997).

User authentication is the process whereby it is confirmed that an entity involved in a transaction is who it claims to be at that point in time. Data origin authentication, on the other hand, is used to prove the source of a particular message. Certificates based upon cryptographic operations can be used to identify assets and endpoints in a Cloud environment, thereby validating an identity. A digital signature can also be used to perform data origin authentication, since the generation of such signatures require knowledge of a cryptographic key.

Data integrity is the assurance that the data has not been modified by entities not authorised to do so. Digital signatures and cryptographic checksums such as Message Authentication Codes (MAC) can be generated on messages in order to achieve data integrity. Such techniques make use of a key which is known only to authorised entities, and any changes in the message would render such signatures or checksums invalid. MACs provide integrity detection when a shared key algorithm is used, while digital signatures achieve this through the use of a public key algorithm. In the Cloud, for instance, the integrity of forensic data such as log files that can be used during an investigation, can be protected through such cryptographic techniques.

Non-repudiation means that an entity cannot dispute that it performed a transaction or sent a particular message. The message receiver can, therefore, prove to a third party that this message was generated by a particular sender. Cryptographic techniques such as digital signatures can help achieve non-repudiation since these can only be generated over a particular message, by an entity that has knowledge of the secret key. Cryptography can, therefore, also be used to implement measures such as forgery detection.

Confidentiality involves making data accessible only to authorised entities, ensuring it isn’t disclosed to unauthorised parties. Cryptography can be used to achieve confidentiality through the use of encryption, protecting data that is at rest, in transit or in use (refer to Section 4.3). Techniques based upon cryptography can also be used for pseudorandom number generation, which can be used to generate keys used for encryption purposes. Since the focus of this study is on Cloud encryption, the rest of this Chapter will delve deeper into the use of encryption in achieving confidentiality.

4.2 Encryption as a Threat Countermeasure

In Section 3.2, a number of threats which a Cloud risk assessment might identify, were described. In this Section, these are revisited in order to analyse whether encryption and key management practices can help mitigate such risks.
Cloud Encryption and Key Management Considerations

Threats to Confidentiality
This generic type of security threat affects how confidential data is stored and managed in the Cloud. Encryption, accompanied with good key management, can help provide the necessary data confidentiality by ensuring that only users who are in possession of the decryption key can access the plaintext. Encryption can help protect against both malicious and accidental disclosure of confidential information, which can occur as a result of procedural issues as well as software or hardware threats (Cloud Security Alliance, 2010). For instance, if as a result of an attack on the Cloud infrastructure, the data gets stolen or compromised, then encryption would make it considerably difficult for an attacker to extract the plaintext from its encrypted form, ensuring its confidentiality.

Threats to Integrity
These threats result in data being tampered with, or otherwise compromised. As discussed in Section 4.1, rather than encryption, other cryptographic techniques such as digital signatures and forgery detection mechanisms can help verify the integrity of the data in question.

Threats to Availability
Threats to the availability of the Cloud infrastructure, services and data stored within it need to be considered and addressed. Not only is encryption of limited use in this area, but when used it introduces new dependencies upon encryption and key management services, whose availability also needs to be managed. Therefore, when using encryption, it is important to consider techniques such as key escrow as well as the use of standard and interoperable encryption techniques, making it possible to successfully decrypt the data at all times, even when the primary encryption services are unavailable.

Virtually Unlimited Compute Power
The fact that the Cloud infrastructure is flexible and can provide vast amounts of compute power and resources upon demand, makes it an attractive platform for an attacker to use when launching an attack. Encryption will not address this threat directly, since encryption operates on the data and has nothing to do with the actual Cloud infrastructure. Other cryptographic techniques may, however, help indirectly to control the use of the Cloud resources, avoiding unauthorised use of the Cloud infrastructure. For instance, a digital signature which is generated over a random challenge and whose verification might be facilitated through certificates, can be used to ascertain a user’s identity on the Cloud without disclosing the secret key. This makes it easier to identify malicious users, thus preventing the Cloud’s resources from being abused.

Cloud Management
The services and APIs which are exposed over the network to enable Cloud management need to be protected to avoid unauthorised use. Threats to confidentiality, integrity, availability and authentication of these management interfaces need to be considered. Encryption can help to partially address this threat since it makes it considerably harder to spoof or manipulate such requests or messages (Security as a Service Working Group, 2011). By encrypting the messages, confidentiality can be guaranteed, ensuring that an eavesdropper cannot decrypt the Cloud management communications. Encryption will not provide integrity, but as discussed in Section 4.1, this can be guaranteed through the use of cryptographic techniques such as digital signatures, which also protects against message forgery. Encrypted timestamps can also be used to ensure message freshness, protecting against replay of Cloud management messages.

Cloud Provider Insiders
Employees at the Cloud provider might have access to the data stored on the Cloud infrastructure, for management and support purposes. However, such rights can be easily abused by malicious users. Data encryption can be one way of addressing this threat, by making this
information meaningless to the provider unless the correct key is known, protecting against insiders who manage to obtain unauthorised access to the data (Security as a Service Working Group, 2011) (Cloud Security Alliance, 2010). However, encryption on its own is not sufficient to address such a threat, since it needs to be accompanied by good key management practices. If the provider has, or can easily obtain, access to the decryption key, then the data can be easily decrypted by the provider, circumventing the data security provided through encryption. Data at rest should, therefore, be encrypted to protect against insiders. However, depending on the level of trust that the user has in the provider, it might also be appropriate to encrypt data that is in transit or in use in order to address this particular threat. This is because the provider might have ways of monitoring and logging data that is being transmitted or processed, and enciphering it makes it more complicated for the provider to access the actual data. Moreover, any actions performed on the data by the provider, including access for management or support, should be logged and audited and these log files should be protected using cryptographic techniques.

Shared Cloud Infrastructure
Since the Cloud infrastructure is shared between multiple tenants, it is possible for a user to exploit a vulnerability in the hypervisor in order to gain malicious control over the Cloud infrastructure. Data encryption is only of limited use against this threat since confidentiality is not the main concern. This threat can be addressed using other cryptographic techniques such as code signing, which can be used to allow the provider to verify that the software running on the Cloud infrastructure is from a trustworthy source and that it has not been tampered with. This is essential since it helps avoid running software on the Cloud that might have been maliciously modified to introduce new software vulnerabilities. Code signing on its own, however, does not guarantee that the software does not contain any software vulnerability, but it helps ensure that the software was not modified once it was released by the manufacturer.

Data Security
Threats to data on the Cloud include deletion, modification and loss of confidentiality. Encryption of data at rest, in transit or in use can help ensure the data’s confidentiality, but it cannot protect against the other threats. Encryption can also protect against network traffic intercepted through man in the middle attacks or sniffing, since encrypting data in transit means that this will not be sent as clear text (Cloud Security Alliance, 2010). Encryption algorithms used should be strong and well-established, and keys should be of sufficient strength depending upon the nature of the data that is being encrypted. Good key management practices should also be in place to ensure that the keys involved in encryption and decryption are not compromised, as this would in turn compromise the security of the data in question.

Access to Cloud Data
Authentication of Cloud users is important since otherwise an attacker may obtain unauthorised access to the user’s data or Cloud management rights. The attacker might also use this information to then launch further attacks. Authentication mechanisms, access control and authorization techniques should be in place, and cryptographic techniques such as digital signatures can be used to facilitate identity validation and to avoid malicious modifications to access control data, both by insiders and by external attackers. Encryption, however, is of limited importance in this area.

Unknown Risk Profile
The fact that the Cloud consumer trusts the provider to manage the Cloud infrastructure is, in itself, a possible security threat since the provider is blindly trusted to ensure that all necessary security measures are taken. The Cloud consumer might not be aware of how risks are being assessed and managed, to what extent, and what technologies or solutions are being used to do so. It is, therefore, sometimes desirable for the Cloud consumer to also take some precautionary
measures, to strengthen any security already provided by the Cloud provider. Encryption is one such example, where rather than relying only on the provider’s measures to keep the data secure, the consumer can also encrypt the data before transmitting it to the provider. Key management can also be addressed in a similar way, whereby techniques such as split keys can be used (refer to Section 6.2.1), effectively sharing the key management responsibility between the provider and the consumer.

Forensics
Security-critical actions and operations on sensitive data should be audited. This applies equally to actions done by the provider as well as by the consumer. As discussed in Section 4.1, cryptographic techniques can be used to protect the integrity of this data and provide non-repudiation. Encryption, on the other hand, can help ensure that its confidentiality is maintained and that privacy issues are managed, by ensuring that access to this audit data is restricted to a need to know basis.

Threats to Compliance
Threats to compliance with legislation and standards might emerge as a result of the risk assessment process. Failure to achieve compliance with laws such as the Data Protection Act and the Safe Harbour Agreement, to mention some examples, can have serious consequences for the organisation. Depending on the legislation in question, approaches such as encryption can be used to achieve compliance by ensuring that data confidentiality is maintained (Security as a Service Working Group, 2011) (Cloud Security Alliance, 2010). Whilst some laws explicitly state that encryption is required, others indicate that the use of encryption can help achieve compliance even when unexpected events take place, such as the loss of confidential data. Section 5.3 provides further details on how encryption can help achieve compliance with various types of legislation.

Encryption, therefore, has a role to play in a Cloud environment, in order to address particular information security issues. A risk assessment should be performed using CC-RAM and, depending on the findings of the particular scenario being assessed, the applicability of encryption as a possible countermeasure should then be considered. Moreover, apart from directly addressing particular security threats, the use of Cloud encryption typically introduces greater peace of mind and reduces the level of perceived risk, making the Cloud a more attractive prospect.

4.3 Where to Encrypt Data

As discussed in Section 3.2, the Cloud is subject to a number of threats which can be identified using a risk assessment technique such as CC-RAM (refer to Section 3.4). Section 4.1 and Section 4.2 proceeded to discuss the use of cryptography in general and the suitability of encryption, respectively, in order to address particular problems with information security in the Cloud. However, despite the fact that encryption can be used to help ensure data confidentiality, its implementation is not straightforward and requires much thought and consideration. Amongst these decisions, one needs to determine at which point of its life cycle the data should be encrypted and decrypted. Failure to encrypt data at the required point of its lifecycle would mean that this might be unnecessarily compromised at its most vulnerable state. On the other hand, applying encryption to data when it is not required would result in an unnecessary overhead in terms of complexity, computation and performance.

Data can be in one of three main stages - at rest, in transit or in use (Cloud Security Alliance, 2012). Data that is at rest is stored in a database, in other structured or unstructured files, or
objects on disk. In the Cloud, this type of data is stored on the Cloud infrastructure by the provider. Data that is in transit, on the other hand, involves data that is being transmitted between the sender and the intended recipient. In the Cloud, this might be between two endpoint computers connected together via the Cloud, or also between two different internal applications or services of the Cloud (such as between the Cloud application and the backend Cloud database). Data that is in use is that which is currently being processed, used or displayed by the consumer. Encryption can be applied to data that is in one or more of these states, the suitability of which depends upon the particular scenario.

**Encryption of Data at Rest**

This encryption provides confidentiality for data that is currently stored. It, however, raises a number of challenges on how encryption can be done without losing the ability to perform certain operations on the data, such as reading of indices or performing a search on the main fields of the object. Encryption of data that is at rest can be done using techniques such as “Digital Rights Management (DRM) schemes, native file system encryption, and application software development tools. Hard drives can be encrypted with full disk encryption implementations available as an OS feature as well as self-encrypting solutions following the Trusted Computing Group’s OPAL standard” (Cloud Security Alliance, 2012). Data that is at rest can be encrypted by the provider or else directly by the consumer before sending it to the Cloud, rather than relying on the Cloud provider to provide this security. This approach helps ensure that the data will definitely be stored in an encrypted format. The fact that the provider does not know the encryption key means that the data is potentially more secure, but this also limits the operations that the provider can do with the data (Cloud Security Alliance, 2009). When performing encryption of data that is at rest in the Cloud, advanced encryption techniques can be used, that store the encrypted data split across several locations of the Cloud to make sure that the data remains secure even if a repository is compromised.

Encryption of data that is at rest is frequently used in IaaS where, for instance, the volume is encrypted. This is also achievable in PaaS which, however, might involve some additional complexity depending on the data storage mechanisms used. In SaaS, this type of data encryption can be used only if it is inbuilt into the Cloud application’s logic, since the logic to encrypt the data as it is being stored needs to be implemented directly within the SaaS application’s code.

**Encryption of Data in Transit**

Established protocols such as Secure Socket Layer (SSLv3) and Transport Layer Security (TLS) can be used to transmit data over a secure channel that provides confidentiality through encryption. Such techniques can be used to protect data that is being transmitted over a network or which is in transit in the Cloud (Cloud Security Alliance, 2012). SSLv3 and TLS are well established protocols that make use of a symmetric key that is distributed using asymmetric cryptography, in order to encrypt the data (Internet Engineering Task Force, 2011). Alternatively, the Cloud can make use of other techniques, the choice of which depends on the type of protocol being used to transmit the data between the two endpoints. For instance, IPSec can be used to provide a secure channel between the endpoints, encrypting the data that is being transmitted over the tunnel. Alternatively, Secure / Multipurpose Internet Mail Extensions (S/MIME) can be used to encrypt data that is being sent via email (Cloud Security Alliance, 2012).

Encryption of data in transit can usually be achieved with similar complexity in all the Cloud models – namely SaaS, PaaS and IaaS (Cloud Security Alliance, 2009). In SaaS, encryption of data in transit can be accomplished by a third party proxy sitting between the consumer and the provider, ensuring that the provider only receives, uses and stores encrypted data. Encryption of data in transit, however, is particularly more important in public or hybrid Cloud deployment models as opposed to private Clouds. This is because when the private Cloud is hosted on
premise at the Cloud consumer, then the data would only need to be transmitted over the organisation’s internal network, which contrasts with data being transmitted in a public Cloud over the internet. However, ultimately the decision on whether data that is in transit should be encrypted must be based upon the findings of a risk assessment and on the organisation’s risk appetite.

Encryption of Data in Use
Data that is currently in use might be exposed by the Cloud as a result of vulnerabilities or flaws in the infrastructure. For instance, vulnerabilities such as core dumps of the Cloud infrastructure’s memory might result in the unencrypted version of the data currently in use, being disclosed. Therefore, encryption of this data is desirable in order to ensure that this is not compromised. However, this gives rise to a number of problems when attempting to process the data, since this processing usually needs to be done on the original plaintext rather than the ciphertext. If encrypted data is going to be loaded and processed in memory, usually this first needs to be decrypted before it can be used and displayed. This makes it difficult for data that is in use to be encrypted at all times (Cloud Security Alliance, 2012). It is, therefore, desirable to be able to perform meaningful processing on data that is still encrypted, without the need to know the corresponding decryption key.

The Cloud deployment model plays an important role in determining the importance of encryption of data that is in use. A public Cloud, for instance, is accessible over the internet and shared with many tenants. This increases the chances of having an attacker being able to access data that is being processed, as a result of vulnerabilities in the Cloud infrastructure. On the other hand, in a private model access to the Cloud infrastructure is more controlled and restricted, as is its use, making it less important to use encryption to ensure confidentiality of data that is currently in use.

Self-Protecting Data
Apart from encrypting data that is at rest, in transit and in use, the concept of self-protecting data has also been put forward, which ensures that the data is maintained secure and encrypted at all times, without the need of having to handle this encryption explicitly whenever performing an operation on the data (Cloud Security Alliance, 2012). In self-protecting data, the data consists of two parts – an encrypted blob, and metadata which describes the blob. Details stored in the metadata include who can access the data, what operations can be performed on it and also the data’s type. Whilst this approach is desirable, there still is no real-world implementation of this which is sufficiently refined and standardised, and it remains more of a theoretical concept rather than an actual implementation (Cloud Security Alliance, 2012).

Encryption of Keys
Irrespective of which type of encryption is considered, key management also needs to be taken into consideration (Cloud Security Alliance, 2009). If the key is managed exclusively by the consumer, then the Cloud provider will not need to be involved in this process. However, if the provider is involved in the data encryption processes, then it needs access to the cryptographic key. All encryption keys should be treated as sensitive data, and should be secured while in storage, transit and use, possibly being encrypted using higher level keys. Moreover, access to these keys should be controlled and restricted to particular entities that can perform specific operations on the data. The keys also need to be backed up in order to ensure that if these are lost, then they can still be retrieved, allowing the encrypted data to be decoded. There are a number of standards and guidelines which can help guide key management in the Cloud. For instance, OASIS Key Management Interoperability Protocol (OASIS, 2010) is a standard for interoperable key management in the Cloud, while IEEE 1619.3 (IEEE, 2008) discusses key
Cloud encryption and key management considerations in a way that is particularly relevant to IaaS. These are discussed in more detail in Chapter 6.

4.4 Encryption in Different Cloud Architectures

Cloud encryption is a complex task that can be accomplished in a number of different ways, enabling data to remain confidential even when the consumer loses control over this by placing it under the Cloud provider’s care. Irrespective of which approach to encryption is selected, the three main components involved in the cryptosystem are the data, encryption engine and key management. These components can be placed in different parts of the Cloud, depending on the Cloud model being used and the required security levels.

Figure 6 shows the different Cloud service models mentioned in Section 2.3, as concentric circles with IaaS using only a subset of the Cloud’s functionality and infrastructure, while SaaS encompasses most of the Cloud’s offering. In this figure, the different levels of security and varying degrees of control by the Cloud consumer are shown for each service model. As can be seen, SaaS provides the least risks since this architecture is already setup with minimal end-user involvement. On the other hand, in IaaS the Cloud consumer has more control over the infrastructure and also on the security measures to be implemented. In this Section, the three Cloud service models are reviewed in more detail, and the applicability of encryption in each type of model is discussed.

4.4.1 Encryption in SaaS

Security measures in SaaS will typically be implemented directly by the provider (Byrne, 2011), although the Cloud consumer can sometimes also take some additional security measures before sending the data to the Cloud. When considering encryption in SaaS, there are three main approaches that can be adopted (Mogull, 2012):

- Provider encryption after data transmittal
- Consumer encryption before data transmittal
- Third party proxy encryption during data transmittal
Cloud Encryption and Key Management Considerations

Provider Encryption After Data Transmittal
In this approach, encryption is performed by the Cloud provider after it receives the plaintext data from the consumer. The provider can then store or retransmit this data in an encrypted state. In this way, data would be protected against any threats to confidentiality, which the multi-tenant, network-based Cloud infrastructure brings around with it. The transmission of the plaintext data, however, needs to be also secured. A secure channel such as SSL/TLS can be setup between the consumer and the provider to protect the data. However, the fact that encryption is a slow and expensive process means that providers typically do not make use of encryption for all data that is stored and received, and instead chose to focus more on other physical or logical controls that prevent unauthorised access to the data. Moreover, the fact that the same provider is involved in both the encryption and the storage of the data, introduces security concerns due to lack of separation of duties (refer to Section 6.7).

Consumer Encryption Before Data Transmittal
Data can also be encrypted by the consumer before this is sent to the Cloud, using an encryption process that is not linked to the Cloud. The encrypted data is then sent to the Cloud and decrypted again when it is retrieved. In such cases, therefore, the encryption process and the keys are managed by the consumer, although third parties might be involved in certain processes such as the provision of key management services. This approach, therefore, places the Cloud consumer in complete control of SaaS encryption, but means that the SaaS application itself can only perform limited operations (such as searching) on the data. It is, however, not always possible to encrypt all the data before sending it to the Cloud, since this depends upon how the SaaS application is developed. For instance, in a Cloud email application, if the email recipients are encrypted before sending the email to the provider, then the Cloud email application will be unable to deliver this email to the intended recipients. Without knowledge of the decryption key, the SaaS application will not be encryption-aware and will, therefore, be unable to interpret or distinguish encrypted from unencrypted data. For instance, in the Cloud email application, the body of the email might be encrypted such that only the intended recipient who owns the decryption key can then decrypt it, with the SaaS application transferring a seemingly meaningless string of characters between the sender and the recipient. In consumer encryption, therefore, the consumer does not need to rely on the provider to ensure the data’s confidentiality.

Third Party Proxy Encryption During Data Transmittal
A third party proxy can also be used to encrypt the data as it is being transferred between the Cloud consumer and the provider. In SaaS offerings, a network-based encryption proxy application that sits between the consumer and the provider, can automatically encrypt the data being sent to the provider and also decrypt any data received from it. This is completely transparent to the user, and it removes the burden of encryption and key management from the consumer, whilst addressing the separation of duties issue that provider-side encryption introduces. The proxy, however, needs to be trusted and security considerations such as availability and data integrity also need to be addressed.

From the conceptual models described in Section 2.6, Model A is based upon a public SaaS platform. In this model, provider encryption can help ensure that data is kept confidential even after an attack upon the Cloud provider infrastructure. However, this encryption does not protect against insider threats by the Cloud provider itself. Model A should, therefore, use this type of encryption when the data processed by the application is of limited sensitivity. Consumer encryption, on the other hand, can be used when the data processed by this conceptual model is more confidential. By encrypting the data before placing it on the Cloud, its confidentiality will be ensured, possibly at the expense of more limited functionality (such as sorting or searching) by the SaaS application. Such an example would be encrypting a file before uploading it to Dropbox. The definition of Model A also describes that a third party proxy may be used to manipulate the
data as it is transmitted between the consumer and the provider. In this model, a proxy enables the use of encryption for popular SaaS applications that do not have this functionality inbuilt into them. For instance, a proxy may be used to encrypt emails being sent through Gmail. All three options, therefore, are valid for conceptual Model A, depending on the sensitivity of the data and the functionality provided by the SaaS application.

Conceptual Model B, on the other hand, consists of a hybrid SaaS environment which has complicated data flows. Data which is not sensitive is sent to the provider, who is responsible for processing and storing it. Consumer-side encryption of this data would make it difficult for the SaaS application to process it and, therefore, provider-side encryption is instead preferred for the public component of this conceptual model. Model B also assumes that sensitive data is stored at the consumer and, therefore, the consumer needs to handle encryption of confidential data that is at rest. However, this data ultimately still needs to be decrypted and sent from the private portion of the hybrid Cloud to the SaaS application, which can then process and use the confidential data. While this confidential data is not stored on the public Cloud, it needs to be transmitted there so that it can be used by the application. Any consumer encryption on this data needs to be reversed before transmission, unless the SaaS application has sufficient information to be able to directly decrypt the data. The use of a third party encryption proxy which encrypts all the data as soon as it leaves the private Cloud and decrypts it as it is retrieved by the organisation, can also be considered in this hybrid model. This ensures that any data, irrespective of whether or not it is confidential, is encrypted before it reaches the provider. The limitation of this approach is that such proxies are usually only available for popular SaaS applications, and the fact that the SaaS application receives encrypted data can possibly limit the usefulness of the Cloud application. Conceptual Model B, therefore, would ideally use a combination of provider and consumer encryption to secure the data.

4.4.2 Encryption in PaaS

In PaaS, security measures such as encryption can be performed by the Cloud consumer or the provider, or possibly both. There are two main approaches to PaaS encryption:

- Application-level encryption
- Provider encryption at the Cloud infrastructure level

**Application-Level Encryption**

In PaaS, an application developer typically writes software that consumes various PaaS services in order to provide the required functionality. These services might include file storage or database hosting, to mention some examples. While developing these applications, the developer usually implements PaaS encryption directly at the application layer (Byrne, 2011). For example, software developers typically write applications to encrypt (or hash) sensitive data such as passwords before this is written to the database. The data is then decrypted again by the application whenever this is retrieved from the database. The advantage of this approach is that the developers are in complete control of the encryption process, and can manage this as required. Moreover, the data is never sent to the service provider in its plaintext form. However, this approach requires custom code to be written and incorporated into the application, something which can be complex to do securely and slow to execute. Other Cloud services (such as Encryption as a Service) might be called directly by the application to outsource the encryption process back to the Cloud.
Cloud Encryption and Key Management Considerations

Provider Encryption at the Cloud Infrastructure Level

In PaaS, the provider is will typically be responsible for ensuring that data stored on the Cloud infrastructure is not compromised. This is achieved through the use of low level measures which are transparent to the consumer. In the previous example of a database service, the provider might opt to encrypt the database files stored on disk, preventing unauthorised users from opening these files should they somehow manage to obtain a copy. Compared to application-level encryption, this approach is more generic and usually covers the whole database rather than specific fields. However, this decryption only takes place once the data arrives at the provider after being sent by the application consuming the PaaS services. Another disadvantage of this system is that the principle of separation of duties is not enforced, with the provider being responsible for both data storage and encryption.

Conceptual Model C, which is a public PaaS system, stands to gain from both types of encryption. Application-level encryption provides the consumer with more control over the encryption and key management processes, ensuring that the data is kept confidential even from the provider. On the other hand, low level encryption performed by the provider can help protect the database as a whole from malicious third parties, a process which may be compute intensive but which the Cloud infrastructure can handle. In Model C, therefore, both forms of encryption can be implemented simultaneously, since in PaaS encryption and security are a shared responsibility between the Cloud consumer and the provider, where the consumer takes care of application-level security and the provider is responsible for lower level controls.

4.4.3 Encryption in IaaS

Security in IaaS is a shared responsibility between the provider and the Cloud consumer (Byrne, 2011). This is because whilst responsibility for data security typically lies with the consumer, the provider is then responsible for securing the infrastructure itself using measures such as firewalls and server load balancing. When considering data encryption in particular, in IaaS this is mainly the responsibility of the Cloud consumer, since this decides what data is to be processed on the Cloud and how its security will be managed throughout its entire lifecycle.

Encryption in an IaaS model can be performed in a number of ways (Kamaraju, 2011) (Mogull, 2011):

- Volume-based encryption
- File-based encryption
- Application-level encryption

Volume-Based Encryption

In volume-based encryption, the data is encrypted directly at the storage layer, and it is impossible to read the data from disk unless the necessary encryption keys are provided. This ensures that the data is kept confidential and that users, both internal and external to the provider, cannot access this data if the disk is not already mounted using the correct keys. Typically the encrypted volume will be separate from the boot volume, to ease the configuration and management processes. In this approach, therefore, encryption is implemented directly on the Cloud infrastructure, and the encryption engine and key management are shifted to the Cloud.

When considering volume-based encryption, an important decision that needs to be done is where the encryption key should be stored. One approach is to place the key within the volume itself and protect it using symmetric or asymmetric key cryptography. This makes it easy to move
the volume since the key moves with it, but it also means that the key is kept together with the encrypted data, which could be a security risk if an attacker manages to gain access to the volume. For additional security, therefore, the keys can be separated from the encryption engine by using an external key management server to manage them (Mogull, 2011). Such an approach makes it less likely for an attacker to be able to decrypt the data if he manages to take a copy of the encrypted volume, since the provider would then only keep the key temporarily in memory rather than having this written on the volume itself.

**File-Based Encryption**

File-based encryption is also possible in IaaS environments, with each file being encrypted directly by the Cloud infrastructure before this is written to disk. This approach is flexible, since it allows the user to determine which files need to be encrypted, and what algorithm and key-strength should be used. However, this approach only works with files and it cannot be used directly for data that is written to structured storage such as a database.

**Application-Level Encryption**

Another approach to encryption in IaaS is to implement this directly into the applications that will be running on the IaaS Cloud infrastructure, providing the same benefits and drawbacks as in PaaS. This type of encryption is possible in situations where IaaS is used to provide an infrastructure upon which applications which have inbuilt encryption will be executed. In the case of private or hybrid Clouds, external systems or specialised hardware can also be used to perform encryption.

Conceptual Model D described in Section 2.6, consists of a private IaaS Cloud deployment which assumes that the whole Cloud infrastructure is stored on premise at the consumer. For this reason, as discussed in Section 3.2, the threats faced in this model are significantly less than its public counterpart. However, measures to ensure confidentiality of sensitive data might still be appropriate, depending upon the findings of a risk assessment. Being a private model, volume-based encryption might not necessarily be a priority, since the volume itself will be protected by organisational security measures and controls. Whilst the use of this type of encryption is not excluded, it is typically not essential and the additional computational overheads might not be justifiable. However, approaches such as file-based encryption and application-level encryption can be useful in this conceptual model. This is because these types of encryption are more selective, ensuring that only data that really needs to be protected is encrypted. In Model D, it is also appropriate to reuse any existing enterprise key management systems or hardware security modules that the organisation already has in place, in order to manage Cloud encryption in the private IaaS platform.

On the other hand, conceptual Model E is a public IaaS system. Volume-based encryption is often relevant in this model, since the volume is subject to a significant number of risks as it is located on the public Cloud. Whilst encryption of the volume is performed by the Cloud provider, in this model it is ideal for the Cloud consumer to handle the key management aspect, or else outsource this to a third party in order to separate the key from the encrypted volume. In doing so, it becomes less likely for the volume to be decrypted by an unauthorised user. File-based and application-level encryption can also be useful in Model E, depending on the particular application of IaaS that is being evaluated. In fact, through these two types, encryption in IaaS becomes similar to what would normally be done in a traditional on-premise infrastructure.

IaaS, therefore, allows for significant flexibility in the adoption of encryption, since it enables the Cloud consumer to implement this as required. However, this flexibility also introduces higher levels of risk, since it increases the probability of misconfiguring the encryption system or selecting an approach which is not ideal for the given situation.
4.5 Cloud Encryption Trends

As shown in Figure 7, in a survey carried out by Ponemon Institute\(^1\) (Ponemon Institute, 2012), it was found that various organisation perform the data encryption process at different points of the Cloud. From the organisations which were involved in this study and which encrypt data in the Cloud, 38% perform this encryption while it is in transit over the network to the Cloud. This can be achieved using some form of network-based encryption proxy. Whilst ensuring that the data is encrypted before it even reaches the Cloud, it however limits the usefulness of this data since this cannot be processed by the Cloud without the provider having access to the decryption keys. On the other hand, 35% of respondents encrypt the data before transmitting it, while only 27% use the encryption services provided by the Cloud infrastructure. Despite its convenience, generally speaking it is not considered to be best practice to make use of encryption services provided by the same Cloud provider where data is stored, since this breaks the separation of duties principle (refer to Section 6.7 for more details). Nonetheless, a consumer might opt to make use of such Cloud encryption services due to their convenience and usefulness, in which case the consumer should seek to ensure that the provider is following encryption and key management best practices in order to provide a secure and reliable encryption service.

In the same study, it was also found that out of the respondents which use the encryption services provided by the Cloud infrastructure, 74% believe that encryption is mainly the provider’s responsibility. This contrasts significantly with those respondents who do not use Cloud encryption services, where only 34% believe that the provider is the entity who is mainly responsible for protecting the data. These findings highlight the fact that there is a market for both approaches to Cloud encryption – performing encryption client side, and performing encryption at the provider’s end.

When it comes to key management, Figure 8 shows that less than half of the respondents involve the Cloud service provider in key management, while 36% state that their organisation is mainly responsible for key management. Organisations, therefore, are typically willing to relinquish control of their encryption keys to have these securely managed by a third party. The study also found that this is the case even when encryption is not performed by the Cloud provider. By passing on the keys to a secure third party, the organisation can ensure better availability of these keys, such as through key escrow. Moreover, third party organisations that specialise in encryption and key management are more likely to have the necessary technology and policies to

---

\(^1\) Refer to Section 3.3.1 for details about the survey methodology used in this study.
ensure good key management, when compare to organisations that are not specialised in this area. When passing on the keys to a third party, best practices suggest that the Cloud consumer should separate the key management process from the Cloud provider that stores and processes the data, seeing that this separation of roles can help avoid conflicts of interest and other security issues (Cloud Security Alliance, 2009). However, in cases where the provider is responsible for key management, it is in the consumer’s interest to know what key management policies the provider implements, including what key generation, storage, usage and disposal procedures are used (Cloud Security Alliance, 2009).

When considering Cloud encryption that is performed by the service provider, the cost in terms of performance and compute resources must also be considered. This is particularly important given that the same resources on the Cloud infrastructure are shared between multiple tenants. For this reason, providers will typically opt to keep encryption to a minimum, sometimes encrypting only particular data fields such as passwords, or else not encrypting anything at all (Granneman, 2011). Other providers opt for some form of obfuscation, which doesn’t provide true security since it is merely security through obscurity, something which can be bypassed more easily than encryption. Unfortunately, in encryption algorithms there is a trade-off between security or functionality, and performance. An active area of research in Cloud Computing is, therefore, the development of encryption algorithms which are suitable for the Cloud, and which consume the least resources possible.

4.6 Conclusion

Some of the security issues of Cloud Computing can be addressed through the use of cryptographic techniques. In particular, encryption is often a suitable measure for ensuring confidentiality of data placed on the Cloud, since it prevents the data from being disclosed to unauthorised users. In fact, encryption and good key management practices can be effectively used to address, at least in part, a number of Cloud-specific threats. Encryption, however, does not come without its challenges and difficulties. One of these challenges is deciding where encryption should be applied, in order to ensure that it is effective, efficient and not unnecessarily limiting. The various Cloud service and deployment models, for instance, all contribute an important role in the decision of whether data is to be encrypted while at rest, in transit or even while in use. The fact that in the Cloud there usually are multiple actors, including the consumer and provider, makes it more difficult to implement and manage encryption in a reliable, secure and consistent way. Therefore, whilst encryption is a useful measure for ensuring information
security in the Cloud, as discussed further in Chapter 5, its implementation and management can be difficult and challenging.
Challenges of Cloud Encryption

As it is frequently said, encryption is not a panacea that solves all security issues, but it might be suitable to address particular security concerns such as confidentiality. When considering security controls like encryption, one must first understand the challenges that will be faced in the process of implementing them, in order to better guide their implementation. This will be discussed in Section 5.1 of the study. Section 5.2, on the other hand, will review the challenges involved in key management, which is a critical component of any encryption system. It is important for such issues to be analysed and understood, in order to ensure the security of the cryptosystem. Legal and compliance issues also need to be understood when dealing with the Cloud, something which is particularly difficult since, owing to its geographically-dispersed nature, the Cloud might cross several legal jurisdictions. An insight into this will be provided in Section 5.3. On the other hand, Section 5.4 identifies some standards related to Cloud Computing, and analyses what role encryption plays in achieving these standards.

5.1 Encryption Challenges

Encryption is sometimes considered to be the ideal countermeasure that can be taken to prevent data security issues in the Cloud. Whilst encryption does have its own advantages in addressing particular security issues, this technique also introduces its own set of challenges that should be addressed in order to ensure that it not only achieves its security aims, but that it also does not prove to be a hindrance to the Cloud users.

The challenges of Cloud encryption are quite similar to those of encryption solutions in general, with the added complexity that the Cloud introduces (Cloud Security Alliance, 2012). These can be classified into two categories – those challenges which are inherently present in the encryption process, and those challenges which need to be addressed when implementing desirable (but non-essential) functionality in an encryption system. As discussed in this Section, the nature of these challenges varies according to the Cloud architecture being used (refer to Section 4.4).

5.1.1 Inherent Challenges

Encryption Policies
Differences in the encryption policies of the various parties involved in Cloud encryption, make the management of encryption significantly challenging. In fact, the management and alignment of encryption policies becomes increasingly difficult as more entities are involved, for example when a third party is used to perform key management. Although it is possible to address this challenge by having the provider perform all encryption-related operations, this would then however go against the principle of separation of duties which states that the entity storing the data should ideally not be also responsible for key management. Therefore, the challenge is to manage encryption policies such that the threats discussed in Section 3.2 are addressed, including threats to Cloud management, threats created by Cloud provider insiders, and also threats to data security.

Availability of Encrypted Data
If Cloud data encryption is used, then ensuring the availability of this data can be a challenge and this is particularly true if this encryption process is done by the Cloud provider. This is because if the provider’s cryptography services go down or the service is deteriorated, then the data owner might be unable to access the data in its unencrypted form, unless alternate key stores and decryption services are made available. A similar situation occurs if the provider loses the cryptographic keys, which is a threat against availability (refer to Section 3.2).
Cloud Encryption and Key Management Considerations

Integrity of Encrypted Data
Another challenge introduced by data encryption in the Cloud is how to ensure that the data is not tampered with once it has been encrypted (Anthes, 2010). Although simply modifying parts of an encrypted file will probably result in garbage data once this is decrypted, it is desirable to have techniques in place that allow the user to identify when the encrypted data has been tampered with. Note, however, that it is also possible for an attacker to overwrite a file’s contents with valid ciphertext that would have been read from another encrypted file. As discussed in Section 3.2, the fact that the Cloud infrastructure is shared and that there is an unknown risk profile involved, makes the issue of data integrity an important one in the Cloud.

Encryption Client Security
When considering Cloud encryption, one needs to not only consider the security aspects of the Cloud infrastructure, but also those of the actual client. The Cloud is often accessed from a variety of devices, including computers and mobiles, and not all of these have the same levels of security inbuilt into them. If the client’s security is compromised, then when confidential data is being transmitted to or from the Cloud, the session between the Cloud consumer and the provider might potentially be hijacked by a malicious user. As discussed in Section 3.2, this hijacking would result in this data’s security being put at risk (Cloud Security Alliance, 2012).

Compliance with Legislation and Standards
Not all challenges introduced by Cloud encryption are of a technological nature. In fact, as discussed in Section 3.2, legal and compliance issues related to the storage of data and to its encryption also need to be addressed when dealing with encryption, something which is complicated due to “scant legal or regulatory framework, and few precedents, to deal with issues of liability among the parties in Cloud arrangements” (Anthes, 2010).

Key Management
When considering any encryption process, one must also consider the related key management issues and how these can be addressed in order to ensure that the keys used do not get compromised. This is especially important if the keys are stored on mobile devices that are used to access the Cloud, since these devices are prone to getting lost, thus compromising the security of these keys. When addressing key management issues, the entire life cycle of the keys needs to be considered, right up to their destruction once the data they protect is not required any longer (Cloud Security Alliance, 2012). Key management and its challenges are reviewed in more detail in Section 5.2.

5.1.2 Challenges in Implementing Desirable Functionality

Processing Encrypted Data
In order to be able to work with encrypted data, this usually first needs to be decrypted before it can be used and processed. Decryption is a heavy operation which is expensive both in terms of time and processing power required, so alternatives that allow data to be used in its encrypted form without compromising its confidentiality, are desirable.

Searching Encrypted Data
It is not uncommon for users to access their data through search, and this is sometimes the only feasible approach when there are large amounts of data to navigate through. This, however, becomes significantly more complicated when the data that needs to be searched for is encrypted since, unless the decryption key is known to the Cloud infrastructure, this data cannot be easily matched with the search terms (Anthes, 2010). Moreover, even if the key is known to the Cloud infrastructure, it would still be too computationally expensive to constantly have to decrypt the
Cloud Encryption and Key Management Considerations

Data in order to search it. For this reason, searching within encrypted data proves to be a challenge.

**Sorting Encrypted Data**

Similar to searching, sorting is yet another challenge that is introduced when the data is encrypted. Unless a technological approach is used to address this issue, the only solution to sorting encrypted data would be to first decrypt it. This is inefficient and also introduces security risks since the data would be in its plaintext format while being processed.

**Encryption Interoperability**

Once encrypted data is accessed by the consumer, this needs to be decrypted in order to be able to use it. This decryption can take place on various types of devices and environments, making interoperability of key and encryption mechanisms a major consideration in Cloud encryption (Cloud Security Alliance, 2012). Moreover, interoperability is desirable since it enables a Cloud consumer to change Cloud or encryption service providers with relative ease. It also allows for third parties to act as backup agents for performing key management or encryption functions, ensuring better availability of encryption services.

**5.1.3 Cloud Model Analysis**

When analysing the inherent challenges of Cloud encryption together with those challenges encountered while implementing desirable functionality, he different Cloud service or deployment models being used should be taken into consideration. This is important since these models influence the challenges involved. For this reason, this Section reviews these challenges in the light of the conceptual models described in Section 2.6.

The management of encryption policies between the various entities needs to be considered in all Clouds, but it is especially important in hybrid Clouds where the data constantly flows between public and private Clouds. Conceptual Model B, which consists of a hybrid SaaS environment, demonstrates this clearly. The private and public Clouds are each governed by a separate policy and, therefore, the differences between these need to be understood and managed to ensure a consistent and functional policy. This is particularly important given the complex data flows and the distinction between sensitive and non-confidential data. On the other hand, out of all the Cloud deployment models, a private Cloud provides the least challenge to the management of encryption policy. Conceptual Model D, which consists of a private IaaS system, involves the least number of actors since the Cloud is completely managed by the consumer organisation and, therefore, there are fewer different policies involved in this type of model.

On the other hand, the challenge of ensuring the availability of encrypted data is mostly effected by the Cloud service model used. Conceptual Model B consists of a SaaS environment where encryption is performed by the provider, while Model A assumes that this is done by a third-party proxy. In either of these two models, the availability of the decryption services needs to be guaranteed. In contrast, in PaaS, encryption is mostly managed by the software developers. For instance, in conceptual Model C, which is a public PaaS environment, the developer consuming the PaaS services can opt to encrypt the data before placing it in a database or saving it to disk. Decryption is then also performed by the PaaS application being developed. In Model E, which is a public IaaS system, encryption and decryption of volumes is done by the operating system or other software running on the IaaS infrastructure, with the keys being managed by the provider or the consumer. Since the IaaS operating system is controlled by the consumer, this actor has more control over the availability of the encrypted data. Model D, which is a private IaaS system, allows for even more control, since the whole Cloud is managed by the organisation with no third
Cloud Encryption and Key Management Considerations

party involvement. Whilst this challenge of ensuring the availability of the decryption services remains in all models, this responsibility can therefore be shifted from the provider towards the Cloud consumer.

Apart from availability, one also needs to address the integrity of encrypted data. This challenge is particularly important if the data owner does not have direct control over the encryption keys, such as in conceptual Model A, which is a public SaaS application where encryption is managed by the provider or a third party proxy. This concept also applies to Model C, which is a public PaaS system, where low level encryption measures such as database encryption in Database as a Service, may be taken by the provider. From the conceptual models defined in this study, Model D provides the organisation the most control over the integrity of the encrypted data, since this is a private deployment model.

The challenge of ensuring security of the client machine used for encryption is true irrespective of the Cloud model being used. Client machines performing encryption are often potentially insecure since the machines are either exposed over the internet or used for purposes other than simply encryption. In SaaS, for example, applications are often accessed by a client using a web browser. This client may also perform client-side data encryption before uploading the data to the SaaS application, for example when encrypting and uploading a file to Dropbox. This client computer can be vulnerable to attacks, especially if it has not been hardened, and the confidentiality of the data being processed can be compromised as a result.

When considering compliance with legislation, the Cloud deployment model influences the degree of impact that this challenge has on the Cloud consumer. Compliance is considerably more complicated in hybrid or public Clouds, where the data or the different entities are often spread across different countries and jurisdictions, each one having its own set of legislation. The consumer, provider and the various Cloud servers might, in fact, be governed by different laws which are sometimes quite diverse or even conflicting. In conceptual Model A, which is a public SaaS model, once the data is placed on the Cloud, this flows within the Cloud infrastructure so that it can be processed or stored. This infrastructure might not be in the same geographical region as the Cloud consumer. Model C and Model E, a public PaaS and IaaS model respectively, face the same challenges as the public SaaS model. Model B, being a hybrid SaaS model, is also affected by this challenge. However, the fact that sensitive data can be stored on premise at the consumer instead of on the public Cloud helps to partially address this issue. On the other hand, the Cloud consumer has even more control over the data in private Clouds. The fact that in scenarios such as conceptual Model D, which is a private IaaS system, the data never leaves the organisation that owns it, makes compliance with legislation easier to achieve. The challenge of legal compliance is discussed in more detail in Section 5.3.

The ability to process data directly in its encrypted form is yet another challenge that encryption brings. In conceptual Model A and Model B where, upon request by the consumer, encrypted data is processed by the SaaS application that is managed by the Cloud provider, the provider should ideally be able to process the data directly in its encrypted form. This is particularly desirable in Model B, which is a hybrid model, so that the sensitive data can be encrypted before it is sent from the private portion of the Cloud to the SaaS application running at the provider. In this way, the provider would be able to process the data without needing to know how to decrypt it, ensuring that its confidentiality is maintained. On the other hand, the ability to process encrypted data without the need to first decrypt it, is less useful in PaaS and IaaS which implement more low-level encryption. For instance, in Model E, although it may be desirable to use an encrypted IaaS volume without first decrypting it, in reality this would limit the functionality and number of operations that can be performed on the volume. For any complex processing, the storage disk or volume would need to be mounted and decrypted in order to be
able to access and have unrestricted use of all data on the drive. In Model D, which is a private IaaS model, this functionality is less important since there are fewer risks than in a public IaaS model, making low-level volume encryption less critical.

On the other hand, the challenges of searching and sorting encrypted data are common throughout all Cloud service models. For instance, searching for particular data in its encrypted form in SaaS Model A, and looking for data that has been encrypted and stored somewhere on the Cloud in IaaS Model E, both provide a similar challenge. Similarly, sorting is of equal difficulty in all service models. In SaaS, this might be needed at the application-data level, such as sorting a list of emails by subject. In IaaS, on the other hand, sorting might be necessary when attempting to order a list of encrypted files stored on the volume.

Yet another challenge in Cloud encryption is the interoperability of the encryption algorithm used, which is desirable in any service model. In conceptual Model A, for instance, if a proxy is used to perform data encryption when this is being transmitted to the SaaS application, then the use of an interoperable encryption algorithm would allow for the proxy to be replaced with greater ease, as long as these proxies support the same technology. In Model C, on the other hand, the use of interoperable encryption in PaaS would allow the application consuming the PaaS services to be easily ported to another Cloud provider.

Cloud encryption, therefore, introduces various challenges which need to be analysed within the context of the Cloud service and deployment models being used. In doing so, these can be better understood such that they can be addressed using approaches such as those discussed in Chapter 7 of this study.

5.2 Key Management Challenges

Key management is the process whereby the keys in a cryptosystem are managed, including their generation, exchange, use and disposal. This includes various protocols and procedures that aim to keep the keys secure at all times, granting access to them only to authorised entities. However, “key management is the hardest part of cryptography, and often the Achilles’ heel of an otherwise secure system” (Schneier, 1996). Cloud encryption is no exception, and appropriate key management mechanisms must be in place in order to ensure the security of the data encryption processes.

When implementing a key management system, one needs to consider where the encryption keys should be stored. These keys may be stored in one of three main locations (Thiemann, 2012):

- **Placing keys in an enterprise datacentre** – When using an enterprise datacentre to store the keys, these are maintained with high levels of security, ensuring that these are not compromised. Such a system does not carry the risk of relying on a third party that might potentially be compromised, as happened with the RSA SecureID breach in 2011 (The New York Times, 2011).

- **Using SaaS to manage keys** – Keys may alternatively be stored using a SaaS key management solution where the application provider will manage and store the keys. This high dependence upon the provider raises a number of security concerns, including the possibility of having the keys unavailable in case of an outage. Legal concerns also come into the picture, with laws such as the USA Patriot Act obliging the provider to provide the encryption keys to the American Government without previously informing the data owner (Rose, 2011).
• **Using IaaS to manage keys** – A third approach is to use the encryption and key management services provided by IaaS. Like the SaaS option, such a system results in strong reliance upon the provider if the customer decides to allow the provider to manage the keys (Thiemann, 2012). However, some IaaS providers also provide their customers with the opportunity to manage the encryption keys themselves, resulting in better separation of duties. Amazon S3 Storage (Amazon Web Services, 2012) is one such example.

Due to its complexity, key management introduces a number of challenges, including (Pate & Tambay, 2011) (Cloud Security Alliance, 2012):

- Key generation and storage
- Key availability
- Key disposal and expiration
- Key management policy
- Separation of duties
- Key management interoperability

### 5.2.1 Inherent Challenges

**Key Generation and Storage**
Generation of keys used during encryption needs to be done in a secure manner, since if the keys are compromised from the very start, then the security of the whole encryption process would be at risk. As discussed in Section 3.2, this would then result in a threat to data confidentiality. Key management techniques which ensure that the generated keys are strong and securely stored, need to also be in place.

**Key Availability**
Once keys are generated and used to encrypt data, one also needs to ensure that these keys are available whenever needed, since this would otherwise be a threat to availability (refer to Section 3.2). This is particularly important since if the encryption and decryption keys are lost, the data might effectively be rendered inaccessible, resulting in temporary or permanent data loss. High availability is, therefore, one of the essential considerations in any key management system, including that for Cloud encryption.

**Key Disposal and Expiration**
Another challenge associated with key management is the disposal and revocation of keys once these are no longer required. Procedures should be in place, which allow keys to be revoked when particular entities who had access to the key should no longer maintain this access (Cloud Security Alliance, 2012). Similarly, keys might also need to be revoked if they become compromised, preventing them from being used for further cryptographic processes. As with any cryptosystem, keys used for Cloud encryption might also expire after a predefined lifetime, since keys often have associated with them an expiry date to protect against cryptanalysis attacks. This key expiration introduces a number of challenges, such as how to manage this process and what operations should be performed on data that was encrypted with the expired key. Failure to address key disposal challenges will result in threats to data confidentiality and possibly also compliance, as described in Section 3.2.

**Key Management Policy**
Key management processes should be part and parcel of a holistic security policy that is adopted by the Cloud consumer or by the provider (Pate & Tambay, 2011). This is a challenge in itself,
since the policy must be consistent or complementary across the different players involved in Cloud encryption, something which becomes more complicated as the number of entities involved increases.

**Separation of Duties**
When considering Cloud encryption, the security principle of separation of duties should be implemented, whereby key management processes should not be under the Cloud provider’s responsibilities (Cloud Security Alliance, 2012). By having the keys and the data fall under the responsibility of separate entities, it becomes more difficult for an attacker to obtain both a copy of the encrypted data and also the keys required to decrypt the data, since these are not kept together. This is also true in the case of any insider attacks that might originate from within the provider. Moreover, separation of duties also protects against threats to confidentiality, integrity and compliance, and against threats created by having a shared Cloud infrastructure, as discussed in Section 3.2.

**Key Management Interoperability**
Interoperability of key management solutions is yet another consideration, which is of particular importance in the Cloud since several entities might be involved in Cloud encryption and key management processes. Key management interoperability is a challenge since different providers and encryption algorithms typically have different key management requirements. Interoperability helps avoid being permanently tied to a specific key management provider, and it also allows for reuse of existing systems, making Cloud management easier and less prone to security risks.

### 5.2.2 Cloud Model Analysis

Similar to Cloud encryption, the challenges that relate to Cloud key management also need to be considered in the light of the particular Cloud implementation being used, such as the service and deployment model. This section will therefore analyse these key management challenges through the use of the conceptual models defined in Section 2.6.

The challenge of secure key generation and storage applies to any Cloud model used. However, diverse models usually result in different entities being responsible for these operations. In conceptual Model A, being a public SaaS system, the encryption keys are generated and kept by the Cloud provider or by a third party proxy, seeing that the consumer is not involved in the data encryption process. In Model B, which is a hybrid SaaS model, the Cloud consumer can be responsible for key generation and storage, especially for the keys needed for the encryption of sensitive data that is stored on premise. In PaaS and IaaS, on the other hand, key generation and storage is usually mainly managed by the Cloud consumer. In Model C, which assumes an application that runs on a public PaaS infrastructure, application-level encryption keys are managed by the consumer application. Similarly, keys used to encrypt volumes in the public IaaS Model E, are often also generated and kept by the consumer, in order to keep them separate from the volume that is stored by the Cloud provider. In fact, Cloud consumers might keep these keys on premise if they already have the necessary setup, such as an enterprise key management system. This is also necessary in the private IaaS Model D, where the whole infrastructure is managed by the Cloud consumer organisation. Irrespective of which entity is responsible for key generation and storage, this is a challenge that needs to be addressed when considering key management in the Cloud.

Similarly, key availability needs to be addressed irrespective of which Cloud service or deployment model is being used. Mechanisms should be in place to ensure that the keys are available, even in
Cloud Encryption and Key Management Considerations

situations where the primary key management services are distributed across different points in the Cloud. Similarly, the challenge of key expiration and disposal applies in all Cloud models, although it becomes more complicated to manage when the keys are not under the control of the entity that is providing the encryption service. In SaaS Model A, for instance, the Cloud provider not only stores and processes the data, but is also responsible for the encryption and key management processes. This makes key expiration easier to handle, since everything is under the control of a single entity. On the other hand, if the different processes are performed by different parties, such as in Model E where the keys used to encrypt the public IaaS volume can be managed by the Cloud consumer, then better communication and policy enforcement techniques need to be in place to ensure that expired or disposed keys are not used.

When the Cloud consumer is practically the only entity involved in key management, such as in Model D which is a private IaaS system, key management policies are also easier to manage. However, policy becomes more complicated as third parties, including proxies and key escrow services, are introduced into the process. For instance, in Model B, the keys used to encrypt the data in the hybrid SaaS system might need to be shared and distributed between the various actors involved. The key management policy, therefore, covers various entities and it potentially no longer falls under the responsibility of a single organisation, making this more complex to manage.

A further challenge in Cloud key management is separation of duties. Whilst separation of duties is desirable, one needs to keep in mind that involving more entities in encryption and key management introduces more complexity in the processes. This principle of separation of duties poses slightly diverse challenges in the various Cloud service models. In conceptual Model A, encryption is performed by the public SaaS application and, key management would therefore need to be separated from the Cloud provider that stores the data. A third party proxy that is entrusted with key management and encryption can be used, whilst storage will be handled by the provider. On the other hand, in IaaS, mechanisms are necessary to store the volume encryption keys separately from the Cloud infrastructure. In conceptual Model E, being a public IaaS model, these keys can possibly be managed by third parties or by the consumer using an enterprise key management system, while the provider stores the actual encrypted volume. The concept of separation of duties does not necessarily apply to Model D, which is a private IaaS model. The decision on whether or not to outsource key management depends on the sensitivity of the data, the degree of trust that the organisation has in its employees, and the security posture of the organisation.

The challenge of key management interoperability is also influenced by the Cloud deployment model used. For instance, in Model B, which is a hybrid SaaS Cloud, there is a strong interaction between the private and public components of the Cloud infrastructure. In this model, it is possible for the Cloud consumer to encrypt the data in the private Cloud and transmit this to the SaaS application still in encrypted format. Interoperable key management makes it possible for the provider to fetch the key and decrypt the data. In a hybrid Cloud where multiple entities are involved in key management, interoperability therefore becomes more challenging yet necessary as it helps simplify and streamline the key management operations across the various encryption systems. Moreover, when the Cloud consumer chooses to trust third parties to perform key management duties on its behalf, the use of interoperable practices enables the consumer to change with ease the entity that is entrusted to perform key management, whenever necessary.

Although “the Cloud may create new key management challenges, the principles for choosing between the various alternatives remain the same” (Thiemann, 2012) as with any cryptosystem. Key management is an essential component of Cloud encryption, and it should be carefully considered in order to ensure that the encrypted data’s confidentiality is maintained. Based upon
the organisation’s risk appetite and its security requirements, the best key management techniques and practices (refer to Chapter 6) should be selected and adopted by the organisation.

5.3 Legislation and Compliance Challenges

In Cloud Computing, data processing is shifted from the user’s computer and network onto the Cloud infrastructure, with the user rarely being aware or able to control where in the Cloud the data is stored. In fact, in the Cloud, there are no geographical boundaries that limit where the data is held, and this is often spread across the Cloud in different data centres found across the globe. Data may also be replicated in different data centres, in order to enhance efficiency and availability (Navetta, 2009). This situation is even more difficult to manage when surge computing is used, where different Cloud players (such as Google and Amazon) might purchase Cloud capacity from each other and shift data between themselves, when they are experiencing unusually high levels of demand (Berkeley Clouds, 2009). Even though this flow, interchange and replication of data are transparent to the user, they create various legal and compliance issues.

Whilst a detailed analysis of the legal aspect of the Cloud is beyond the scope of this study, this section gives an overview of the main legal challenges that concern encryption.

Data Flow Across Different Jurisdictions
The primary legal issue is related to the transborder flow of data, where different jurisdictions impose different data-related laws (Navetta, 2009). For instance, under the EU Data Protection Directive, it is not possible to transfer personal information outside of the EU, unless these countries provide the same degree of data security to personal information, as that of the EU (European Parliament, 1995). In the Cloud, it is very easy to go against this directive, by simply having the data stored in a Cloud server in a country not approved by the EU. Clearly, measures must be taken to ensure compliance and the protection of personal information at all times.

Liability
Another legal issue to be considered is the liability that an organisation has when it opts to move its data to the Cloud and this is then subjected to a security breach (Navetta, 2009). Whilst legally the organisation is obliged to ensure that the chosen provider has sufficient security measures in place to prevent data security issues, it is often difficult for such an analysis to be conducted in a Cloud environment. This may happen because the provider might be outsourcing its work or it may have entered into an agreement with other providers to share parts of their Cloud infrastructure, as happens in surge computing. In such cases, the organisation might want to consider taking appropriate security measures to ensure that the data remains protected even if the provider fails to reach expectations when managing its security.

5.3.1 Legislation Which Affects Cloud Encryption

When considering the legal aspect of the Cloud, different laws and directives come into play, covering sectors such as financial, health care, government and privacy. To mention some notable examples from the USA and EU, these include (Forsheit, 2009) (Lapinsky, 2011):

- **USA Patriot Act** – Applicable to providers that store or process data in the US, which are obliged to provide any data to the US Government under particular conditions
- **Health Insurance Portability and Accountability Act (HIPAA)** - Applicable to organisations involved in the health care sector which process private health information
- **Children’s Online Privacy Protection Act (COPPA)** - Applicable to online data of children
- **Gramm-Leach-Bliley** - Applicable to financial institutions which process private, personal information
Cloud Encryption and Key Management Considerations

- **Sarbanes-Oxley (SOX)** – Applicable to organisations that have obligations on how to preserve, maintain, retain and dispose of data
- **Federal Information Security Management Act** – Applicable to all systems operated by the US Government or its contractors
- **Data privacy breach notification laws** – Applied by several American States and European countries
- **EU Data Protection Directive** – Applicable to personal data

Complying with these laws and other similar ones, is a difficult task, especially in a Cloud environment. Organisations should find ways of achieving compliance before moving their data to the Cloud, since it is often more difficult and expensive to do so once the organisation is already using the Cloud. An approach that can be considered in order to help achieve compliance is the use of encryption. In fact, most US laws, for instance, do not require the organisation to report the loss of sensitive data if this was appropriately and strongly encrypted. In a dynamic and open environment such as the Cloud, the organisation putting data on the Cloud might use measures such as encryption to ensure that the data remains confidential at all times, even though the organisation would have relinquished some of its control over the data by placing it under the Cloud provider’s care. As discussed in Section 4.3, encryption should be considered for data that is at rest, in transit or currently in use.

By encrypting data, therefore, compliance with legislation becomes easier since this is often regarded as a suitable measure to ensure data confidentiality. Some laws clearly state that an organisation would not be liable if it loses sensitive data which is encrypted. HIPAA, for example, requires organisations to report data breaches to the government and the media, but organisations are exempt from this provision if the data was protected using strong encryption (US Government, 1996). In fact, products such as Google Message Encryption have been developed for the Cloud to ensure that sensitive data is automatically encrypted during transmission and while at rest, in order to avoid issues of noncompliance with legislation such as HIPAA (Google Postini, 2009). With regards to key management, HIPAA also states that “to avoid a breach of the confidential process or key, these decryption tools should be stored on a device or at a location separate from the data they are used to encrypt or decrypt” (US Government, 1996), meaning that the Cloud provider should ideally not be involved in the key management process, achieving separation of duties (refer to Section 5.2).

However, other legislation that does not directly make reference to the use of encryption is often insufficiently clear and not always easy to interpret. One such example is the European Union’s Data Protection Directive, which is implemented by its various member states. Whilst the concept is the same across all member states, different countries provide slightly different implementations of this directive. The UK Data Protection Act, for instance, consists of eight principles, which apply to “any information relating to an identified or identifiable natural person” (European Parliament, 1995). The seventh principle of this act states that personal data shall be protected using all “appropriate technical and organisational measures […] against unauthorised or unlawful processing” (United Kingdom Act of Parliament, 1998), and against potential loss or damage. This suggests that encryption can help achieve compliance with the Data Protection Act, since private data would be protected against confidentiality breaches unless the decryption key is also compromised.

Legislation, however, is often insufficiently clear and open to a certain degree of interpretation. For instance, when one considers the applicability of the Data Protection Act when encryption is used, two possible situations should be considered:
- Personal data that is encrypted is still covered by the Data Protection Act, since it still relates to a particular individual
- Personal data is no longer covered by the Data Protection Act once it is encrypted, since the individual to whom this data pertains, is no longer identifiable from the ciphertext

This is an important consideration since organisations placing personal encrypted data on the Cloud do not want to risk possible non-compliance with any applicable law. Whilst this is a grey area, it seems that the Data Protection Directive must still be adhered to even when personal data is in its encrypted form (Corrales, et al., 2010) (Barnitzke, et al., 2011). Whilst the Data Protection Directive states that “the principles of protection shall not apply to data rendered anonymous in such a way that the data subject is no longer identifiable” (European Parliament, 1995), it also states that “to determine whether a person is identifiable, account should be taken of all the means likely reasonably to be used either by the controller or by any other person to identify the said person” (European Parliament, 1995). Whilst encryption provides an additional layer of protection to the data by converting it into ciphertext, it is possible for an attacker to break the encryption used, using large amounts of Cloud Computing resources or alternative techniques such as social engineering. Although such attacks are not usually attempted by casual attackers, they are theoretically possible and reasonable given a determined attacker with sufficient time and resources. The Data Protection Directive, therefore, still applies if an attacker might potentially be able to somehow decrypt the data and identify the person to whom this belongs, since this law applies if the individual can be identified from the data collected in any direct or indirect way (Barnitzke, et al., 2011).

The Cloud, therefore, is subject to a myriad of laws and directives, which are often unclear or inconsistent. It is in both the consumer’s and the provider’s interest to understand which laws apply to the Cloud, and to take the necessary measures to ensure compliance wherever possible. Cloud encryption is one such measure, which can be used to maintain the confidentiality of the data being processed, by converting it to an unreadable form. Some laws explicitly require encryption of sensitive data, and in the Cloud this needs to be applied to data that is at rest, in transit or in use. Key management practices are also dictated by some laws, which require the separation of duties principle to be enforced. However, the use of encryption in itself should not provide a false sense of security, since non-compliance is indeed possible in particular cases even when the data in question is encrypted, since ultimately “the information content does not change when being encrypted” (Barnitzke, et al., 2011).

5.4 Information Security Standards

Despite the convenience and many advantages that the Cloud offers, this is subject to various security issues (refer to Section 2.5). When moving to the Cloud, a detailed risk assessment should be carried out, possibly using the CC-RAM approach (refer to Section 3.4), in order to identify the risks that the Cloud is subject to, and to also determine the countermeasures necessary to mitigate these risks. Encryption and good key management practices are two approaches that can be used to counter some of these risks and concerns. It is, however, essential for both the Cloud provider and the consumer to consider encryption as part of a wider and holistic approach to achieving Cloud security rather than considering encryption only in isolation, in order to ensure its effectiveness and proper management. Various security standards exist which aim to provide a baseline and guide on how information security can be achieved by the organisation, most of which include the use of encryption in some form or another. Whilst some standards were not developed with Cloud Computing in mind, they nonetheless still apply. On the other hand, other standards which focus on Cloud security are also being developed and updated. Amongst the various security standards that should be considered in the Cloud, one
finds ISO27001, ISO27002, ISO27005, BS25999, COBIT, and the Open Security Architecture, to mention some examples.

5.4.1 Generic Information Security Standards

ISO27001 is a standard which describes a model for “establishing, implementing, operating, monitoring, reviewing, maintaining and improving an information security management system (ISMS)” (International Standards Organisation, 2005), based upon the organisation’s security requirements. Figure 9 shows the basic concept of this standard, which consists of four phases built into what is known as the Plan Do Check Act cycle (International Standards Organisation, 2005). During the Plan phase, the organisation defines its security objectives and establishes the ISMS, selecting the security controls which it considers necessary to counter the identified risks. Encryption and other cryptographic techniques form part of the extensive list of possible controls identified by this standard. During the second phase, the controls previously identified are implemented, and during the subsequent phase these are regularly monitored and reviewed to ensure that they are still effective and sufficient to counter the security risks. When such controls fail to reach their objectives, any improvements necessary are carried out, including the implementation of any corrective or preventive measures to the ISMS.

ISO27002 is a standard which complements ISO27001, by providing “guidelines and general principles for initiating, implementing, maintaining and improving information security management in an organisation” (International Standards Organisation, 2005). ISO27002 provides guidance on implementing security controls which fall under a wide range of information security areas:

- Security policies
- Organisational information security, considering both internal and external parties
- Asset classification and management
- Human resources security, including that prior to employment, during employment, and also after termination
- Physical and environmental security
Cloud Encryption and Key Management Considerations

- Communications and operations management, including network security, media security, security of information being exchanged, and monitoring of security-critical activities
- Access control, including user access rights, network access control, and mobile computing
- Information systems management, with special emphasis on cryptographic controls and file security
- Incident reporting and management
- Business continuity management
- Compliance with legal requirements and standards

In a Cloud environment, a consumer can follow ISO27002 to ensure that the risks of Cloud Computing are mitigated as much as possible. For instance, control A.6.2.1 states that “risks to the organization’s information and information processing facilities from business processes involving external parties” need to be identified (International Standards Organisation, 2005). ISO27002 also requires the Cloud consumer to identify the third parties that critical activities and information rely upon, in order to understand the risks that these entities create. When following this standard, the Cloud consumer should therefore check the provider to ensure that this implements information security practices, and has appropriate processes and procedures in place to address the security risks. If the Cloud provider is certified to this standard, then this also provides assurance to the consumer that the provider is taking the necessary measures to achieve appropriate levels of information security. By choosing a provider that follows such a standard, a consumer can ensure that best practices are followed in the Cloud and that the information is well protected throughout its entire life cycle, using an adequate information security management system. Although being ISO27002 certified, it is still possible for a Cloud provider to be vulnerable to security risks, such as Amazon Web Services which suffered an outage in 2012 (Cook, 2012). However, through ISO27002, potential risks can be lowered by addressing security using a structured approach.

Moreover, control A.6.2.3 states that agreements must be in place to “cover all relevant security requirements” (International Standards Organisation, 2005). In implementing this control, a Cloud consumer should establish specific security agreements with the provider, especially to address high priority risks. Once established, such agreements should also be regularly reviewed to ensure that they are being adhered to by both parties. Through documented agreements, both parties will enter into a legal contract which establishes a common understanding for security measures, in such a way that the consumer knows exactly what security measures to expect from the Cloud provider.

Another aspect of ISO27002 that a Cloud consumer can apply to mitigate security risks, is control A.10.5.1 which requires backups to be taken. The consumer should ensure that the provider takes regular backups, whilst the consumer might opt to also keep a local backup if this is feasible. A plan for data recovery should also be in place.

ISO27002 also refers to the use of encryption in various areas of the standard, and it provides guidelines on the use of cryptographic controls and key management within the organisation. In this regard, ISO27002 states that the use of encryption should be considered for protecting sensitive information, in particular that which is being transmitted through mobile devices or over public networks (International Standards Organisation, 2005). With regards to key management practices, ISO2702 states that keys should be managed securely throughout their entire life cycle, including their generation, distribution, storage, usage, replacement, revocation, archiving and disposal. Usage of the keys should be logged and audited, and recovery should be possible if the
keys get lost, to ensure business continuity (International Standards Organisation, 2005). IS27002, therefore, not only encourages the use of encryption as a possible way of ensuring data confidentiality, but it also provides a set of best practices in order to ensure that encryption and key management are implemented in an effective and secure manner.

Hence, whilst both ISO27001 and ISO27002 are not directly applicable to Cloud Computing, they can both be used as an effective way for both the Cloud consumer and the provider to address information security within the Cloud. Whilst it isn’t easy for such standards to be fully implemented by small organisations opting to move to the Cloud, they provide clear guidance on how any organisation, irrespective of its size, can follow information security best practices. Moreover, standards such as ISO27005 and BS25999 can also be used to support the Cloud provider and consumer in achieving this security. ISO27005 provides guidelines on information security risk management which can be used to successfully implement ISO27001 (International Standards Organisation, 2011). On the other hand BS25999 is a standard on business continuity management, something which is of particular importance to the Cloud (refer to Section 2.5).

Another framework that should be considered when implementing security is the Control Framework for Information and Related Technology (COBIT), which guides governance and control by enabling managers “to bridge the gap between control requirements, technical issues and business risks” (ISACA, 2012). Again, this framework can be applied to the Cloud by using it prior to choosing a Cloud provider and also after the organisation has moved to the Cloud, in order to manage risks, define responsibilities and also ensure separation of duties (Shimba, 2010).

### 5.4.2 Cloud Security Standards

The Open Security Architecture, which works on Cloud security standards, put forward the Cloud Computing Pattern (Open Security Architecture, 2012) which an organisation might consider when implementing Cloud security. As shown in Figure 10, this pattern involves looking at the Cloud from different facets, including from an architect, business user, IT manager, developer, Cloud service provider and end user’s perspective.

Figure 10 shows the various controls that each actor in the Cloud can take in order to ensure security. These controls are classified into a number of categories:

- **Access Control (AC)** – Prevents unauthorised users from accessing the information
- **Audit and Accountability (AU)** – Ensures audit and forensic information is maintained
- **Awareness and Training (AT)** – Caters for the human aspect of information security
- **Certification, Accreditation and Security Assessments (CA)** – Ensures that security is periodically assessed to ensure that it is effective
- **Configuration Management (CM)** – Ensures that configuration changes are managed such that information security is not compromised
- **Contingency Planning (CP)** – Handles situations where the organisation needs to recover its information after a critical situation
- **Identification and Authentication (IA)** – Provides ways for authenticating the entities involved
- **Incident Response (IR)** – Defines procedures for handling information security incidents
- **Personnel Security (PS)** – Caters for employee hiring and termination
- **Planning (PL)** – Ensures that security plans are in place and maintained
- **Risk Assessment (RA)** – Ensures that risks are regularly assessed and handled
- **System and Communications Protection (SC)** – Ensures that information being transmitted or received by the organisation is secured
Cloud Encryption and Key Management Considerations

- **System and Information Integrity (SI)** – Provides ways for ensuring that the information’s integrity is maintained
- **System and Services Acquisition (SA)** – Caters for in-house or third party services, ensuring that they have adequate security measures

The Cloud Computing Pattern, therefore, consists of over 50 security controls which can be implemented by different entities in the Cloud in order to ensure that information security is achieved. Each of these OSA controls links to an ISO27002 and a COBIT control, thereby facilitating the implementation of these standards in a Cloud environment. Various controls in this pattern, such those in the AC, AT, IA, PS, PL, RA and SC categories, help directly or indirectly to...
ensure that data remains confidential. The controls proposed in the Cloud Computing Pattern state what the security goals are, but they do not provide exact details on how they should be implemented. For instance, the CC-RAM approach for risk assessment proposed in Section 3.4 of this study can effectively be used to implement controls RA-03 (Risk Assessment) and RA-04 (Risk Assessment Update) of this pattern, since it provides a structured risk assessment method which is adapted to the Cloud.

Once the risks are identified, the applicability of the various OSA controls should be reviewed. After considering the baseline for the Cloud scenario whose security is being analysed, this pattern involves identifying the gaps in the implemented security compared to the required controls, and any missing measures should then be implemented in order to mitigate the identified risks (Open Security Architecture, 2012). For instance, the Cloud Computing Pattern provides controls for encryption and key management, which can be implemented either by the Cloud provider or else by the consumer organisation itself. Section 4.4, as well as Chapter 6 and Chapter 7 of this study, analyse which entities should be responsible for the encryption and key management processes in the various Cloud models.

Whilst the OSA Cloud Computing Pattern is a valid attempt to provide a standardised approach to information security in the Cloud, unfortunately providers do not always follow security standards or frameworks. Various efforts to provide other standardised approaches to Cloud security are currently being undertaken, such as the Trusted Cloud Initiative (Cloud Security Alliance, 2011) and the Common Assurance Maturity Model (CAMM) (Common Assurance, 2012), to mention some examples. It is hoped that in the near future, such standards will become more widespread and adopted in the Cloud, in order to provide a more secure environment to operate in.

5.5 Conclusion

Whilst encryption can help achieve confidentiality in a Cloud environment, it introduces a number of challenges. The management of encryption policies and ensuring the availability and integrity of encrypted data, are amongst the main challenges presented. Other challenges that must be addressed involve ensuring the security of the encryption client, as well as achieving compliance with legislation and standards. Apart from these inherent challenges, when performing Cloud encryption one needs to find suitable ways of implementing desirable functionality such as searching and sorting of encrypted data. The processing of data directly in its encrypted form, as well as the use of interoperable encryption algorithms are also desirable but challenging tasks. Apart from encryption, one also needs to understand and address the challenges associated with key management. These include the generation, storage, disposal and expiration of encryption keys, which need to be governed by a suitable key management policy. Interoperability and availability of the key management functions also need to be addressed, and the principle of separation of duties should be enforced. It is important that such challenges are addressed since encryption plays an important role in implementing generic information security standards such as ISO27001 as well as Cloud-specific standards such as the Cloud Computing Pattern. Cloud-specific standards are, however, unfortunately still in their infancy and plenty of work still needs to be done in this area.
6  Best Practices and Proposals for Cloud Key Management

In Cloud encryption, key management can be done by the consumer, the provider or a trusted third party (Credant, 2011). The consumer will usually opt to manage the keys when the consumer organisation already has the necessary infrastructure in place to perform encryption and key management. In this approach to key management the organisation, therefore, retains control of the keys and always has a clear picture of when and where the keys are being used. However, such an approach means that unless the keys are shared, no external parties will be able to decrypt this data. In order to do so, the keys would need to be shared outside of the organisation, and more advanced key management techniques might be necessary to ensure that the keys remain protected.

The second approach to key management involves entrusting the Cloud provider with this responsibility. From a consumer point of view, this simplifies the encryption process since there are fewer aspects for the consumer to manage. Such an approach means that the provider can process the data with ease since it can decrypt and encrypt as needed, but this implies that the consumer loses control over the entire key management process. By being in complete control, the provider and its employees would have the technical ability to decrypt the data at will, something which introduces a security risk. Moreover, there is a heavy reliance on the provider, who is trusted to apply adequate key management processes to protect the keys used during encryption. However, due to its simplicity, such an approach might be convenient for sensitive information which is not of great value to the organisation.

A final approach to key management involves relying on a third party to provide Cloud-based key management and encryption services (Credant, 2011). Together with a number of other Cloud-based security services, such an approach is referred to as Security as a Service (SaaS). In this approach, keys are managed by the third party, and these rarely need to be released externally since this third party also provides encryption and decryption services. By entrusting these keys to a third party who is specialised in this area, their exposure is limited and key management is simplified for the consumer. Moreover, this can potentially be more robust that if it were to instead be done by the consumer or by the provider. Such an approach to key management is, therefore, adequate when the encrypted Cloud data needs to be accessed and managed from a variety of platforms or entities, and when the organisation is comfortable with involving a third party in the process (Credant, 2011).

In this Chapter, a number of best practices are discussed, which apply to Cloud key management. The use of key management standards is discussed in Section 6.1, while Section 6.2 discusses approaches to key generation and storage, as well as the use of split key management. Key availability and distribution are reviewed in Section 6.3 and Section 6.4, respectively, with the latter also focussing on Federated Key Management. Section 6.5 discusses key termination, while the importance of key management policies is mentioned in Section 6.6. The Chapter moves on to discuss the principle of separation of duties in Section 6.7, while an overview of key management interoperability is given in Section 6.8 which emphasises the relevance of the Key Management Interoperability Protocol in the Cloud. The Chapter concludes with Section 6.9, where the human aspect of key management is discussed.

6.1  Use of Key Management Standards

Key management is a complex process which, amongst others, involves technological, managerial and procedural decisions what are required to ensure that the keys remain secure at all times. This complexity and reliance on human involvement means that key management is usually the
Achilles’ heel of any encryption system. When considering Cloud encryption, key management best practices should be considered and implemented, in order to make this process as robust as possible. This can be facilitated through the use of established key management standards and protocols. Such techniques typically cover specific aspects of key management, including the storage, availability, retrieval, sharing and revocation of keys. When considering key management standards that somehow apply to the Cloud, most notably one finds NIST SP800-57, ISO 11770-1, IEEE 1619.3 and OASIS KMIP.

**NIST SP800-57 Recommendations for Key Management**

This standard provides a set of recommendations and information about encryption frameworks and key management approaches. Whilst it does not go into detail about specific cryptographic modules, it describes a set of best practices which are mainly associated with the key management aspect of cryptography. This standard consists of three parts, with the first giving background information and a general set of recommendations on key management. Amongst others, these topics include a description of the possible security services and types of keys that can be used in encryption, as well as general information about cryptographic algorithms and key material (Barker, Barker, Burr, Polk, & Smid, 2007). SP800-57 also describes the protection mechanisms that should be applied to the different types of key management information involved. Moreover, it describes the key lifecycle and the various tasks involved in the key management process. Topics such as “key usage, cryptoperiod length, domain parameter validation, public key validation, accountability, audit, key management system survivability, guidance for cryptographic algorithm and key size selection” (Barker, Barker, Burr, Polk, & Smid, 2007) are also discussed in this document. The second part of these guidelines discusses the organisational requirements needed in order to successfully perform key management. These requirements include established security policies and organisational management processes. Finally, the document delves deeper into key management, providing guidance that is specific to particular encryption implementations. Although not specific to the Cloud, such standards can be adapted to the management of Cloud encryption keys.

**ISO 11770-1 Information Technology Security Techniques for Key Management**

This standard provides a generic approach to key management which does not depend upon specific encryption algorithms or techniques. “ISO 11770 addresses both the automated and manual aspects of key management, including outlines of data elements and sequences of operations that are used to obtain key management services. However it does not specify details of protocol exchanges that might be needed” (International Standards Organisation, 2010). This standard, therefore, provides a framework that identifies the main objectives of key management and provides a general model upon which key management processes should be built. It also identifies the main services of key management as well as their characteristics, and it also defines how keys should be managed throughout their entire life cycle. Subsequent parts of this family of standards discuss general mechanisms for more specific types of keys, such as symmetric and asymmetric key mechanisms. ISO 11770 describes an approach to key management, which helps ensure that they keys’ “origin, integrity, timeliness and (in the case of secret keys) confidentiality can be guaranteed to both direct and indirect users” (International Standards Organisation, 2010). The key management processes covered include key generation, storage, sharing, revocation and archiving.

**IEEE 1619.3 Standard for Key Management Infrastructure for Cryptographic Protection of Stored Data**

“This standard specifies an architecture for the key management infrastructure for cryptographic protection of stored data, describing interfaces, methods and algorithms” (IEEE SISWG, 2010). Amongst the aspects covered, this standard discusses how the keys should be stored, managed and shared. Moreover, it builds upon other established key management techniques (such as
Cloud Encryption and Key Management Considerations

previous standards of the IEEE 1619 family), adapting them specifically to the protection of encryption keys.

**OASIS Key Management Interoperability Protocol (KMIP)**
This open standard aims to facilitate the interoperability of key management processes by “standardizing communication between encryption systems that need to consume keys and the key management systems that create and manage those keys” (OASIS, 2012). Such a standard allows organisations to employ encryption and key management processes that are interoperable, facilitating the participation and interaction of different entities in these processes. “By defining a low-level protocol that can be used to request and deliver keys between any key manager and any encryption system, KMIP enables the industry to have any encryption system communicate with any key management system” (OASIS, 2012). This standard is of particular importance to the Cloud, since it enable the consumer, provider and also trusted third parties to have collaborative key management that is performed in a standardised way. Moreover, standardisation reduces the risk of lock-down since, when compared to situations where proprietary or non-interoperable approaches are used, this makes it easier for the consumer to change the provider or the third party that is managing the encryption process or associated keys.

Note that whilst IEEE 1619.3 focusses on high level key management infrastructure, KMIP deals with the key exchange protocol itself. These two standards are, therefore, complementary and in fact the IEEE standard was recently discontinued and parts of it were incorporated into the OASIS KMIP specification (IEEE SISWG, 2012). Due to its importance in the Cloud as the standardisation benefits that KMIP introduces, this shall be discussed in more detail in Section 6.8.1.

### 6.2 Key Generation and Storage

Best practices also suggest that all necessary measures must be taken to protect the keys throughout their entire lifetime. Keys should be generated in an environment that is secure, using adequate techniques which include good random number generators, to ensure that these are truly random and unpredictable. Moreover, keys should be generated for a single-purpose, in order to avoid unnecessarily compromising them by using the same key for different applications (Federal Office for Information Security, 2011). For instance, some public SaaS applications similar to those described by conceptual Model A use different encryption keys for different Cloud consumers when implementing provider encryption. By having keys that are specific for a particular use, security is increased by limiting the key’s exposure and restricting the amount of information that is compromised should the key be disclosed. In conceptual Model D, which is a private IaaS infrastructure, the keys would be generated and stored by the consumer. Again, such keys should be single use, and the consumer should ensure that it has sufficient technological resources to generate strong and random keys. If this is not the case, then such a task should be outsourced to third parties that are equipped to do so.

Moreover, once the keys are generated, they “should never be transmitted in the clear and should be stored inside a secure element, such as a smart card, or a service with the hardware security module” (Cloud Security Alliance, 2012). Such techniques help ensure that the encryption key does not get easily disclosed, by limiting its exposure to the outside world. For instance, in conceptual Model B which is a hybrid SaaS cloud, the decryption key might need to be transmitted to the Cloud provider so that this can decrypt data coming from the private Cloud. Transmission of this key should be done over a secure channel such as SSL, and the key can be protected using a higher level key such as the Cloud provider’s private key in asymmetric key cryptography.
Cloud Encryption and Key Management Considerations

Keys used for encryption should also be stored in a secure manner, possibly protected using higher level keys, a hardware security module, a smart card, or some other type of secure element (Cloud Security Alliance, 2012). Keys should importantly never be stored in the same Cloud location as the data that they are protecting, since it might be easier for an attacker to obtain both the encrypted data and the keys used if these are kept together. For instance, in conceptual Model E, the keys used for encryption of IaaS volumes can be generated and stored by the Cloud consumer to separate this process from the provider. Such keys could be managed securely through the use of an enterprise key management system.

### 6.2.1 Split Key Management

When considering Cloud encryption, an important key management decision that needs to be done is which party will be storing and maintaining the keys. One possible approach is to entrust the Cloud provider with the keys, however as discussed in Section 5.2, this might introduce some security risks by placing too much trust in an entity external to the organisation. Another approach would be to have the keys maintained by the organisation and stored outside of the Cloud, but this would defeat the benefits of moving the data centre services to the Cloud (Musthaler B., 2012). The consumer can also opt to place the keys in the Cloud itself but it is considered to be good practice to keep the keys separate from the encrypted Cloud data. Therefore, the decision of where to store the cryptographic keys requires careful consideration.

One approach that can be used to work around such an issue is the concept of split key management. In this technique, any cryptographic operation requires the use of two keys in order to complete successfully. One key is stored by the Cloud consumer, while the other is managed by the Cloud provider (Musthaler B., 2012). The concept is similar to having a bank safe deposit box, where one key is maintained by the customer while the other is guarded by the bank, to ensure that access to the box’s contents is only possible when both parties combine their efforts. Split key management can be used in various Cloud service models, such as in SaaS or IaaS, when encryption is performed by the provider but the customer wants to retain some control over the keys used (refer to Section 4.4).

Figure 11 shows a commercial implementation for Cloud split key management, as provided by Porticor (Porticor Cloud Security, 2012). In this example, the Cloud provider keeps its key parts in a Virtual Key Management Service which resides in the Cloud, and it will typically make use of different keys for different applications, to avoid exposing a single key unnecessarily through heavy usage. On the other hand, the Cloud consumer has its own master key which it stores within the organisation using some enterprise key management solution or key escrow service. For any application to perform a cryptographic operation on the Cloud data, the correct key held by the Cloud provider as well as the right master key held by the consumer need to be combined together to form a single key that can be used to complete the cryptographic operation. Such a task is performed by a Virtual Appliance that contains the necessary logic to verify and combine the split keys. For instance, if encryption is being used to protect data that is at rest, then this Virtual Appliance would sit in the Cloud between the application and the actual storage devices, immediately encrypting any data before it is written to storage, and decrypting any data that is being retrieved for processing.

In order to combine together the two key parts, the customer’s master key can be used to encrypt the provider’s keys before these are saved by the Virtual Key Management Service, thereby resulting in the actual encryption keys being held by the provider, but with these in turn also being encrypted by a key known only to the consumer. It is, therefore, impossible for the provider to use these keys without the consumer’s intervention (Musthaler B., 2012).
Split key management, therefore, is an approach which can help address the dilemma of where to store the cryptographic keys. From the conceptual models considered in this study, split key management can be effectively used in all models except Model D, which is a private Cloud that is managed completely by the consumer. This is because in this model, the provider and consumer are essentially the same, meaning that split key management would simply introduce complexity but provide little or no advantages. In the other public or hybrid conceptual models considered, split key management can however be effectively used to introduce shared key management responsibility, albeit at substantial technological complexity and processing overhead.

**6.3 Key Availability**

Another key management best practice is to have mechanisms in place which ensure that keys can still be retrieved in the event of loss or damage to the primary key storage service, or to the mechanisms that protect the keys. Key escrow might be a suitable option which ensures that keys used to decrypt the protected data will not get lost, ensuring that these keys will always be available when necessary (Federal Office for Information Security, 2011). Key escrow can ensure high levels of availability by supplying the keys to a trusted third party, who will not release them to anyone other than authorised parties, following appropriate verification.

At a high level, a key escrow system typically consists of three main components (Bishop, 2003):

- **User Security Component** – This can be part of the key escrow system or part of the key distribution system, and it supports encryption by adding a data recovery field to the encrypted data
- **Key Escrow Component** – This manages the data recovery keys, including their storage and use
- **Data Recovery Component** – This retrieves the plaintext from the ciphertext, when used with the User Security Component and the Key Escrow Component

The Key Escrow Component makes use of escrow agents, which are trusted third parties that manage and run the escrow service. This service makes use of data recovery keys, which can be used to perform decryption when needed. This data recovery process as well as the potential release of the keys is only done by the escrow agents in specific situations, after appropriate authorisation to do so is obtained from the key owner. The escrow agents will make use of technical and procedural mechanisms to ensure that the keys are securely stored.
Key escrow, therefore, provides the advantage of ensuring the availability of the data decryption key. Access to the key is granted when necessary and appropriate, although the fact that a third party is involved in the key management process might give rise to privacy and security concerns, including the potential of insider attacks. The fact that the keys are stored by an entity that is an enticing target to an attacker is an additional concern when using key escrow. Moreover, this introduces additional complexity and costs into the key management process (Abelson, et al., 1997).

Key escrow can be used in any Cloud service models. If the keys are owned by the customer, then this might opt for a key escrow service to ensure availability. The hybrid SaaS Cloud described in conceptual Model B and the public IaaS Cloud defined in Model E, are two such examples where the consumer could benefit from the assurances provided by key escrow. Similarly, if the keys are owned by the provider, then this might also decide to escrow the keys to ensure availability. Such a situation might be the case with the public SaaS infrastructure described by Model A, or with the low-level database encryption keys needed by the provider for PaaS encryption in Model C. The decision to escrow the keys, therefore, is up to the key owner, irrespective of which party this is.

6.4 Key Distribution

Apart from key storage, key distribution should also be done securely. Best practices recommend that measures should be taken to ensure the keys’ confidentiality, integrity and authenticity during distribution (Federal Office for Information Security, 2011). This might include splitting the key before distribution, encrypting it with higher level keys, and covering it with a digital certificate, to mention some examples (Piper & Murphy, 2002). Moreover, mechanisms to facilitate the provision of the keys to all those entities that require it and are authorised to obtain a copy, should be in place. As discussed in Section 6.4.1, Federated Key Management is one approach that can be used to address this.

6.4.1 Federated Key Management

Techniques such as Federated Key Management also facilitate the key distribution process by enabling different systems to access the required keys from the same key management server, allowing the different systems to collaborate together in what is referred to as a federated environment (Martin, 2009). For instance, through federation, two applications located in different places, one performing encryption and the other carrying out decryption, can both communicate with the same key management server. Federation, in fact, facilitates this process and reduces overheads that would be required if the different applications were to each have their own key management systems in place.

Standards such as NIST SP800-57 (Barker, Barker, Burr, Polk, & Smid, 2007) and ISO 11770 (International Standards Organsiation, 2010) provide some information on what key management federation is, but they do not describe in detail how this is to be implemented. Protocols such as OASIS KIMP describe an interoperable standard that also supports federation (refer to Section 6.8.1).

Within a Cloud Computing environment, federated key management might be useful when, for instance, encryption is performed by the Cloud consumer and decryption is done by the Cloud provider, in order to allow the data to remain encrypted while it is in transit over the network. Such an example would be in conceptual Model B, which is a hybrid SaaS infrastructure. Sensitive data can be encrypted and stored at the consumer, and then sent to the public SaaS application in
its encrypted format, where it is then decrypted and processed. This ensures that the data remains encrypted while it transit, and federated key management can help the Cloud consumer and provider to share the same key management server.

Another application for federated key management is when different applications and servers within the vast Cloud infrastructure need to perform some form of processing on the data stored in the Cloud. Such an example would be conceptual Model A, which assumes a vast public SaaS application on the public Cloud. As described in this model, such a Cloud infrastructure consists of several servers and has complex data flows within the public Cloud. Key management services might be required by various nodes within the public Cloud, in order to be able to perform encryption or decryption when necessary. Without federation, key management would become more complex and possibly even less secure, since alternative ways to share the keys between the required parties would need to be implemented.

In federated key management, a key is not only identified by its unique identifier but the key also includes details about the federation key management server (Martin, 2009). This concept of extending the key to include supporting information for key management is similar to other approaches such as Identity Based Encryption (refer to Section 7.6.1). Any application which is trusted by the federated key management server and which knows how to handle such extended key information, can then access the key management services provided by this server. Some form of authentication between the clients and the server is, therefore, essential in such a federated environment.

A federated key management system consists of a number of components, the most notable being (Martin, 2010):

- **Client API** – Includes functionality to authenticate with the key management server, as well as an interface to access the key management services offered by it.
- **Key Manager** – Provides the functionality to manage the keys throughout their whole lifecycle, and exposes key management services to the outside world.
- **Key Management Protocol** – Describes the communication protocol and the messages to be used for interaction between the clients and the server. The protocol covers aspects such as key retrieval and key revocation, to mention two common examples.
- **Policy Description Language** – A language is needed to define what access is to be granted to the different entities, and which operations each of these can perform.

Whilst allowing multiple applications, servers and parties to access the same key management system with relative ease, federated key management also introduces some disadvantages. Since the Key Manager is centralised in a federated system, a single point of failure is introduced. If the key manager is unavailable, then all the cryptographic processes which rely upon its services will be unable to execute. Consider a Cloud consumer who is performing client side encryption. If the consumer takes care of its own key management processes rather than relying upon a federated system, then the organisation would have more control upon key management and can therefore take the necessary measures to ensure that any unavailability of this process is kept to a minimum. On the other hand, through federation, the organisation would need to depend upon a central Key Manager which might not be within its control, and which might be an attractive target to attackers.

Moreover, in federated key management, it is necessary for the Key Manager to establish some form of relationship with the clients consuming its services. Such relationships introduce issues of authentication, authorisation and trust, which need to be addressed for federated key
management to be successful. In addition, a secure API to facilitate key transmission and a flexible yet unambiguous Policy Description Language needs to be in place since a vulnerability in these components would result in a weak key management system which would have a negative security impact across the whole federation. These considerations can prove to be particularly important yet challenging to address, especially when the entities participating in the federation come from different organisations, as is often the case in such systems.

Federated key management, therefore, is a possible way of facilitating Cloud key management and distribution by centralising the key management process and having different entities within the federation consuming these services. However, despite its advantages, it should be considered carefully since federation gives rise to concerns which need to be addressed in order to ensure that security issues are not introduced at the expense of convenience.

6.5 Key Termination

Once the initial keys are distributed, a way of changing existing keys and distributing new ones should also be established. This should be done regularly, the frequency of which depends upon the encryption and key management techniques being used, the sensitivity of the data being protected, and the security policy of the organisation.

Keys that are past their lifetime should be archived securely or destroyed in a secure manner, ensuring that these are unrecoverable (Federal Office for Information Security, 2011). Techniques such as data translation might be required in situations where the key needs to be terminated but the data it protects is still required. This, however, should be carefully considered since this translation is a computationally expensive process.

6.6 Key Management Policy

When considering key management best practices, the organisation should establish an appropriate key management policy before moving to the Cloud (Cloud Security Alliance, 2012). For this to be effective, however, it should be inbuilt into a complete policy framework adopted by the organisation, allowing the organisation to better understand the ultimate goals that key management is trying to achieve rather than simply looking at the narrow picture. Through an effective policy, the organisation can ensure that key management processes are better understood and easier to follow by everyone involved in the process.

Various standards and recommended procedures exist to guide the organisation in defining this policy, such as SP800-57 (Barker, Barker, Burr, Polk, & Smid, 2007) and SP800-131A (Roginsky & Barker, 2011) by NIST, which focus on key management and cryptography respectively. By establishing key management policies, organisations can ensure that their key management practices are conformant with standards and best practices.

6.7 Separation of Duties

A best practice in key management is what is known as the principle of separation of duties. This principle is broken when key management and data storage are both performed by the same entity, such as the Cloud provider (refer to Figure 12). Such an example would be conceptual Model E, if the provider is entrusted to manage the keys required for volume encryption in the public IaaS infrastructure. In such cases, this can lead to a number of difficulties, such as an additional financial burden on the provider as well as potential threats to the data’s security.
Cloud Encryption and Key Management Considerations

Maintaining the keys and the encrypted content together at the provider, is the simplest yet possibly most insecure form of key management.

![Figure 12 - Cloud Key Management Service](Image)

Source: [Cloud Security Alliance, 2012]

Moreover, in cases where the provider is also responsible for maintaining the keys, this provider might not be sufficiently equipped to do so in the best way possible. In fact, key management requires expensive hardware security modules or similar security mechanisms to ensure that the keys are kept secure [Cloud Security Alliance, 2012]. In cases where the provider does not already have such mechanisms in place, then it might decide to implement cheaper alternatives to the hardware security modules, which often provide weaker security measures thus leaving the keys vulnerable.

When the principle of separation of duties in key management is introduced, these keys may be managed by the customer. This is especially viable if the customer already has enterprise key management systems in place. This separation is also relevant in most public or hybrid Clouds. Figure 13 shows such a situation, where although encryption is done by the Cloud provider, the keys are managed mainly by the customer or a trusted third party. The provider makes secure calls to the customer’s key management server in order to obtain the correct key when this is required. For instance, this can be used in public IaaS infrastructures such as in conceptual Model E, or even SaaS setups such as Model B.

![Figure 13 - Remote Key Management Service](Image)

Source: [Cloud Security Alliance, 2012]

Figure 14 shows a slightly different situation, where like key management, the encryption process is also decoupled from the provider. In such cases the provider does not require any keys, and is simply responsible for storing and returning encrypted data when this is requested. Such an approach is more suitable for a SaaS environment such as Model A, where data is sent to the application directly in an encrypted format, allowing the provider to not necessarily be encryption-aware.
6.8 Key Management Interoperability

Interoperability of key management is desirable since this allows for a single key management system to be used by multiple entities and applications. Techniques such as the Key Management Interoperability Protocol (KMIP) which is issued by the Organization for the Advancement of Structured Information Standards (OASIS), provide standard ways to achieve interoperability in key management. The use of standard key management techniques should, therefore, be considered to be one of key management’s best practices.

6.8.1 OASIS Key Management Interoperability Protocol

OASIS Key Management Interoperability Protocol (KMIP) is an emerging standard for achieving interoperable key management, which specifies the communication protocol that is to be used between an encryption process and a key management system. Since the various entities and systems which somehow interact with key management systems do so in a standard way when this interoperable protocol is used, the key management system becomes simpler to manage. This is because KMIP makes it possible to reuse existing key management processes and architectures. Interoperable protocols also make it less likely for the customer to get permanently tied to specific providers or third parties that provide key management services. KMIP was first issued in 2010, and since then has grown and evolved, incorporating other standards and recommendations such as IEEE 1619.3. This interoperable key management protocol has also been considered by the OASIS KMIP Technical Committee as a possible and viable approach to standardising key management in the Cloud (OASIS, 2012).

In any key management system, various aspects need to be managed, such as symmetric keys, public keys, private keys and digital certificates, to mention some examples. The encryption algorithm that is used determines what type of data will be handled during key management. Organisations typically have several applications or systems that make use of encryption, and whose keys need to be managed. As a result, such organisations typically find themselves having multiple key management systems that need to be handled, as shown in Figure 15. Whilst this approach means that the various key management systems can be adapted to the particular encryption system which they are related to, this situation is clearly not scalable or maintainable. Moreover, changing an encryption system would probably also necessitate a new key management system due to lack of interoperability. KMIP tackles this situation by providing a single protocol that is used for all communication between any key management system and encryption system, as shown in Figure 16.
Cloud Encryption and Key Management Considerations

Figure 15 - Proliferation of Key Management Systems
Source: (OASIS, 2009)

Figure 16 - Key Management with KMIP
Source: (OASIS, 2009)
In KMIP, a single Enterprise Key Management system is used to communicate with all encryption systems, through a model consisting of objects, operations and attributes (OASIS, 2009).

- **Objects** – These are the key information that needs to be managed, such as symmetric and asymmetric keys, digital certificates and other user-defined information. These different objects will be accessed through KMIP in a standardised, interoperable way.

- **Attributes** – These are the properties that define the object. Base object properties define core aspects such as key length, while extended attributes are used for storing user-defined information.

- **Operations** – These are the actions that will be performed on the objects, such as creation, retrieval, updating or deletion. By calling various operations, the keys will transition from one state to another, such as Pre-Active, Active, Compromised, Deactivated, Destroyed or Destroyed Compromised (OASIS, 2009).

Figure 17 illustrates the main objects, attributes and operations supported in KMIP, and the reader is referred to (OASIS, 2009) for a detailed account of each. In essence, KMIP works by converting a message from an encryption system into a KMIP request to the Enterprise Key Management server, through the use of objects, attributes and operations. The server will then reply with a KMIP response containing the requested objects (OASIS, 2009). KMIP, however, does not provide an actual API for this message communication, but it instead simply defines the communication protocol that is to be implemented by systems that support this interoperable protocol. The clients will typically initiate requests to the key management server, and then receive and process the KMIP response obtained. It is, therefore, essential that the encryption system processing the response is familiar with the KMIP objects, attributes and operations involved.

Figure 17 - KMIP Operations, Objects and Attributes
Source: (OASIS, 2009)
Since the aim is for KMIP to establish itself as the leading open standard for key interoperability, this protocol was in turn based upon other standards and established protocols (OASIS, 2009). The key life-cycle used in KMIP is taken from NIST SP800-57, while authenticated secure communication between the key management server and the encryption system is achieved through SSL/TLS and HTTPS. KMIP also assumes the use of standard encryption algorithms and key generation techniques. In doing so, KMIP not only follows established best practices for key management, but it facilitates its adoption by organisations that already adhere to such standards.

Therefore, “by enabling support for interoperability between cryptographic clients and enterprise key management systems, KMIP reduces infrastructure costs and the risks in adopting cryptographic solutions as an essential element of securing information, identities and infrastructure” (OASIS, 2009). However, KMIP does not only apply to traditional computing environments, but it can also be successfully used in the Cloud. In such an environment, the Key Management Server (KMS) as well as the encryption system can be located in different parts of the Cloud infrastructure. Four possible such configurations are (OASIS, 2012):

- The KMS and the encryption system are both located at the Cloud consumer
- The KMS is located at the Cloud consumer while the encryption system is held at the Cloud provider
- The KMS and the encryption system are both located at the Cloud provider
- The KMS is located at a trusted third party while the encryption system is held at the Cloud provider

**KMS and Encryption System at Cloud Consumer**

In this scenario, both the KMS and encryption system that communicates with it use the cryptographic keys, are under the direct control of the organisation using the Cloud. Such a scenario is useful in situations where the Cloud provider does not perform any encryption and this is done directly by the consumer before sending the data to the Cloud. Situations where this can be used include cases such as conceptual Model B, where data encryption is performed by the provider. Through KMIP, the organisation would have adopted a key management system that provides interoperability, allowing it to easily change the encryption system or the location of the KMS without having to worry about interfacing with a new key management system.

**KMS at Cloud Consumer, Encryption System at Cloud Provider**

As shown in Figure 18, in this scenario the KMS is still under the responsibility of the organisation, but the encryption services are provided by the Cloud provider. For this reason, the Cloud provider needs to interface with the KMS at the organisation in order to source key management services, a process that is achieved using the Cloud provider’s Key Management Client (KMC). As before, key management is still under the organisation’s control, but the interoperability introduced by KMIP means that the application making use of these key management services can be easily moved to different areas within the Cloud. A typical example of this is in conceptual Model E, where the key used to encrypt the public IaaS volume is managed by the consumer.
Cloud Encryption and Key Management Considerations

KMS and Encryption System at Cloud Provider
In this scenario, depicted in Figure 19, the KMS and the encryption system (together with the KMC) are both shifted into the Cloud, under the provider’s responsibility. This scenario is typical in SaaS applications which use encryption that is managed by the provider, such as in conceptual Model A. In this particular example, a separate KMS within the Cloud is created for each consumer making use of these services, reducing the risks of having the key management services abused as a result of Cloud multi-tenancy. By moving both the KMS and the KMC to the Cloud provider, the organisation would have shifted most of the key management responsibilities to this entity. It is, however, also possible for the Cloud consumer to optionally have a KMS within the organisation which will communicate with the provider’s KMS and replicate some or all of the behaviour. This replication increases performance since, if required, the organisation can use the KMS services without having to go to the Cloud provider. Moreover, this increases the availability of the key management services, since there would be two KMSs providing similar functionality.
KMS at Trusted Third Party, Encryption System at Cloud Provider

In the final scenario for KMIP adoption in the Cloud, the KMS is shifted to a trusted third party who provides key management services while the encryption system and KMC are maintained at the Cloud provider, as shown in Figure 20. In this scenario, the KMS is a shared service provided by a specialised key management provider in the Cloud. As in the previous scenario, the Cloud provider and the Cloud consumer both have the possibility to replicate the KMS services locally, with these in turn interfacing with the KMS at the key management provider. This replication increases performance and availability.

KMIP, therefore, is of significant importance to key management within a Cloud encryption system. By introducing interoperability, all Cloud players avoid the need of having specialised key management systems for particular encryption system implementations. Moreover, the Cloud consumer can avoid becoming too dependent on a particular Cloud provider or third party, since such services can be easily provided by any entity which supports KMIP. The fact that KMIP is used also allows a certain degree of flexibility by making it easier to shift the KMC or KMS in different parts of the Cloud, without needing to worry about interoperability.

Despite the advantages, the fact that KMIP provides interoperability by standardising the communication protocols for key management, introduces some disadvantages. This is because techniques that do not follow standardised protocols might ultimately be more flexible, expressive or efficient than an interoperable one such as KMIP, since they can be designed to address a particular scenario rather than being generic. Moreover, the KMIP client used must be able to interface with the encryption system used, something which might not be easily achievable if using specialised encryption algorithms. In the Cloud, this lack of flexibility or expressiveness can be limiting since it might result in unsuitable key management operations being done by the Cloud consumer or provider. KMIP, therefore, introduces a trade-off between interoperability and functionality.

In addition, KMIP requires additional layers and components in the key management system, introducing further technical complexity and overheads. This might not only have a negative effect on performance (Tysowski & Hasan, 2011), but this complexity can also introduce vulnerabilities or weaknesses in the key management process. It is important that this potential
security issue is addressed, especially in a Cloud environment which is distributed, accessible over the network and supports multi-tenancy (refer to Section 3.2 for further information about these Cloud threats). KMIP, moreover, does not leverage sufficiently upon the scalable potential of the Cloud. For instance, in SaaS it is very easy for Cloud resources to be added or removed according to demand and the number of users. However, if KMIP is being used, then the number of KMSs might also need to grow or decrease according to the demand for key management services, to avoid KMIP being a bottleneck. These KMSs need to be setup, and replication between them must also be configured. Therefore, unlike the Cloud infrastructure that it supports, KMIP does not easily scale up automatically.

Hence, as with any solution KMIP introduces a number of advantages and disadvantages which need to be considered when analysing its suitability for a particular Cloud encryption deployment. However, if properly managed, KMIP can provide interoperability which makes Cloud key management easier.

6.9 Human Aspect of Key Management

Key management best practices also recommend that access to the keys and key management functions should be protected and restricted. In the Cloud, the provider and its employees should ideally not have access to the consumer’s keys, in order avoid unnecessary exposure of these keys which could ultimately compromise them. If this is not possible, access to the keys should be restricted as much as possible, ensuring separation of duties between those who have access to the keys, and the Cloud infrastructure administrators (Federal Office for Information Security, 2011). A quorum approach should also be considered, whereby at least two entities are required to access sensitive key and key management functions (Cloud Security Alliance, 2012). Moreover, in order to carry out any key management functions, Cloud provider employees should be first strongly authenticated and actions should be audited to avoid or deter malicious access and abuse of these management functions. Employees involved in Cloud encryption should also be appropriately trained, to ensure that they are aware of the security implications of Cloud encryption and that they have the necessary knowledge to manage this securely. Any key management processes performed should be audited and logged for reference purposes, allowing the organisation to ensure compliance with the security and key management policies adopted (Musthaler L., 2012). Moreover, such audits should be protected to avoid any malicious changes from being done to them.

6.10 Conclusion

Key management is an essential component of Cloud encryption and, therefore, all necessary measures should be taken to ensure that this is implemented in a secure and robust manner. This can be facilitated by following Cloud key management best practices, which aim to address or avoid various Cloud security threats. In the forefront of such best practices, one finds the use of key management standards and the establishment of organisational policies. Moreover, best practices that ensure secure key generation, storage, distribution, availability and termination should also be followed. Other recommendations include compliance with the principle of separation of duties and the use of interoperable key management. In order to implement such best practices, a combination of procedural and technological measures is often required. Amongst such technologies one finds Federated Key Management and the Key Management Interoperability Protocol which, although they have their challenges and disadvantages, if implemented with care they can be effective techniques that help follow the recommended key management best practices.
7 Best Practices and Proposals for Cloud Data Encryption

Encryption is a difficult process that needs to be considered carefully, especially within the complexity of a Cloud Computing environment. Moreover, it is important that any security measures such as encryption do not prove to be too limiting to the end user, as otherwise these might be used incorrectly or not used at all. Various techniques and approaches to encryption exist, some of which have been applied to the area of Cloud Computing only recently. Whilst a comprehensive review of all possible encryption techniques is beyond the scope of this study, this Chapter describes some of the approaches which are of particular interest to the Cloud, and how these can help follow best practices related to Cloud encryption.

Section 7.1 discusses the establishment of an encryption policy as a best practice, which needs to be supported with appropriate technological and procedural controls. Best practices which relate to the encryption algorithms are discussed in Section 7.2, while Section 7.3 focuses on the importance of ensuring the integrity of encrypted data. Client side security is reviewed in Section 7.4, and Section 7.5 is about the best practice of ensuring compliance with existing legislation. The use of encryption techniques which can help facilitate the management of encryption are discussed in Section 7.6, which also gives as examples an overview of Identity Based Encryption and Attribute Based Encryption within a Cloud environment. Encryption interoperability is mentioned in Section 7.7. Section 7.8 discusses how it is considered to be best practice to implement functionality that prevents encryption from being too limiting from a user’s perspective, with the ability to process encrypted data directly, as well as the possibility of searching and sorting it being discussed as concrete examples. Unfortunately, the technology to support this desirable functionality is still an active area of research, with techniques such as Fully Homomorphic Encryption (refer to Section 7.8.1) not yet ready for production use.

7.1 Encryption Policy

An important best practice for Cloud encryption is to have a security policy in place, which treats data in the Cloud with the same level of security as data that is retained by the organisation. Organisations that use the Cloud to store data need to ensure that their “policies and the accompanying enforcement measures extend to cover data as it moves to and from Cloud providers to ensure that the use of encryption continues to meet all mandates for data protection” (Cloud Security Alliance, 2012). If the Cloud consumer is not managing the encryption process, then this should seek to obtain all the relevant information about how Cloud encryption is being performed, including the strength of the algorithm used and key management schemes employed, in order to be able to assess the level of security provided by encryption (Badger, Time, Patt-Corner, & Voas, 2012). Risk assessments should be included to “understand the value of the data it is about to move to the Cloud, determine its risk appetite, and create policy and requirements that it expects the Cloud provider to meet. Encryption is one of the most important and necessary controls for protection of valued data” (Cloud Security Alliance, 2012).

As discussed in Section 4.3, data in the Cloud can be at rest, in transit or in use. Encryption should be applied to one or more of these areas, and the decision depends upon the risk treatment plan obtained following a Cloud risk assessment. Not all data needs the same level of protection, so although encryption might be mandatory for certain data, it might be considered to be optional in other cases. Best practices suggest that access to data that is at rest should be restricted even if this is encrypted, and access should be “granted on a need-to-know basis” (Cloud Security Alliance, 2012). Policies should be in place to govern this, together with enforced access control procedures.
The encryption policy needs to be supported using appropriate technological and procedural controls. From a technical perspective, it is recommended that the encryption system should also enforce integrity checks. The feasibility of using hashing as opposed to encryption in situations such as password storage should also be considered, since hashing has less computation and management overheads than encryption (Cloud Security Alliance, 2012). Sensitive data should also be protected while it is in transit, and it is considered best practice to make use of established secure communication protocols such as IPSec, SSL/TLS and VPN to transfer this data over a secure tunnel (Jamil & Zaki, 2011). Transfer of data between the customer and the Cloud provider, between different components of the Cloud infrastructure, and between the customer (or provider) and third party entities, should all be considered. Conceptual Model A, as discussed in Section 2.6, provides an explanation of all of the flows that are involved. Mutual authentication should also be used where necessary, since this ensures that data is being sent to the correct individuals.

Moreover, the appropriate technical measures should be in place to support disaster recovery procedures related to data encryption. These measures should be taken by the entity responsible for performing encryption, irrespective of which actor this is. For instance, in the public SaaS conceptual Model A, encryption can be performed by the Cloud provider or a third party proxy, while in the private IaaS Model D, this is performed by the Cloud consumer. These actors should have clear disaster recovery procedures in place that ensure that encrypted data can still be decrypted if the availability of the primary cryptography services is compromised. Redundancy of the decryption service is, therefore, considered to be a best practice since this ensures that data availability is not placed at risk if this is encrypted.

The security of sensitive data that is in use or currently being processed by either the Cloud provider or the consumer should be ensured using the appropriate technology and policy. Moreover, the consumer needs to ascertain that the provider fulfils its obligations in maintaining the confidentiality of the data.

### 7.2 Encryption Algorithms

Various types of encryption algorithms exist, which can be broadly split into two main types that are symmetric and asymmetric. Each of these categories has its own particular advantages and issues.

**Symmetric Key Encryption**

In symmetric key encryption, the same key is used to encrypt and decrypt messages. Typical algorithms include AES (Advanced Encryption Standard) and Triple DES (Data Encryption Standard). This type of encryption is simpler, faster and uses less compute resources when compared to asymmetric encryption since it makes use of simpler algorithms and less complex mathematical operations. Moreover, in symmetric encryption it is easily possible to use different keys for different messages, such that if a key is compromised, then only the confidentiality of a specific set of messages will be at risk. However, key exchange is complex in this type of encryption, since this key needs to be set up in advance. Moreover, key management can become overwhelming as this approach to encryption means that different keys will need to be established across different pairs or groups of entities. It is also difficult to determine the origin of a message, since the cryptographic key is shared with possibly multiple users.

**Asymmetric Key Encryption**

Asymmetric key encryption makes use of two keys - a public and a private one. The public key is disclosed to all users, and this is used to encrypt the plaintext. On the other hand, the private key
Cloud Encryption and Key Management Considerations

is kept secret by its owner, as this is used to decrypt the encrypted messages. RSA is a typical asymmetric key encryption algorithm. Asymmetric key encryption simplifies the key distribution problem, since a key is either public or private, but unlike symmetric key encryption it is never shared between a closed group of users. Message authentication, tampering detection and non-repudiation can also be achieved using digital signatures which are built upon similar technology. Despite the simplified key management, asymmetric key encryption introduces other complexities in the process, since mechanisms such as digital certificates are needed to verify the owner of a particular public key. Asymmetric encryption is slower than its symmetric counterpart, since it uses more compute resources. Moreover, if a private key is compromised, then the confidentiality of all messages encrypted with the corresponding public key will be at risk.

In the Cloud, it is common to use a combination of both techniques, whereby symmetric key encryption is used to protect the data and asymmetric key encryption is used to protect the symmetric key. Irrespective of the approach selected, organisations should however always use established encryption algorithms, which are well known and evaluated by security experts for any possible vulnerability. Proprietary encryption algorithms should be avoided, as these might not have been scrutinised sufficiently by experts in the field, making them insecure and vulnerable to attacks (Cloud Security Alliance, 2012).

7.3 Integrity of Encrypted Data

Another best practice for data encryption is to implement mechanisms that enable the Cloud consumer to verify the integrity of the ciphertext, in order to ensure that the data has not been tampered with. This is particularly important in public models such as conceptual Model E, which is a public IaaS infrastructure, and the public PaaS Model C. This is because, as discussed in Section 2.4, public deployment models introduce greater information security risks.

One approach to achieving this integrity issue is to encode the data in a particular format before encrypting it. Any modifications to the ciphertext will be noticeable since upon decryption, the obtained plaintext will not have the expected encoding. This approach does not require a public key infrastructure to be setup, but the entities that will be encrypting and decrypting need to decide on a fixed encoding structure in advance.

Another approach that can be used is a digital signature over the ciphertext or plaintext, which can help ensure that the data was not modified by a malicious user (Cloud Security Alliance, 2012). This is because when the data is being decrypted, the signature is verified to check the data’s integrity and this will be found to be invalid if the data was tampered with.

The Proof of Storage (PoS) protocol designed by Microsoft is another approach that can be used to ensure the integrity of encrypted data, and this is built on public key cryptography (Ateniese, Kamara, & Katz, 2009). In this protocol, a Cloud consumer can make use of a proof of data possession to verify whether the Cloud provider has modified the data. The advantage of the PoS protocol is that this does not require the customer to keep a local copy of the data to perform this check, and this can be done without retrieving the data from the Cloud since most of the processing is done directly by the Cloud provider.

The PoS protocol consists of four main operations (Ateniese, Kamara, & Katz, 2009):

- **Generate** – This is used to initialise the system, generating a public and a private key pair
- **Encode** – This is a probabilistic algorithm that encodes the data using the private key
Cloud Encryption and Key Management Considerations

- **Prove** – This operation makes use of the public key to generate a value known as a proof, based upon a particular encoded file
- **Verify** – Using this deterministic operation, the public key is combined with the proof to determine whether or not the file’s integrity was compromised after it was encoded

Since encryption provides confidentiality but not integrity, appropriate mechanisms must therefore be in place to ensure that the data is not tampered with once it is placed on the Cloud. This measure must be equally applied to both plaintext data as well as to Cloud data that has been encrypted.

### 7.4 Encryption Client Security

In the Cloud, encryption typically requires the involvement of both client-side and server-side components. Whilst the main focus is typically placed on the provider’s infrastructure when considering encryption, it is important to also secure the client devices involved in the encryption process (Cloud Security Alliance, 2012). Such a task should not be underestimated since not all types of client devices, especially some mobile devices, were built with security in mind. Appropriate authentication mechanisms must also be in place to ensure that the user is properly authenticated when accessing the Cloud. Moreover, mobile devices are easily lost, so measures must be taken to protect the cryptographic process from being compromised in the case of such events.

Whenever the client has the necessary resources to perform client side encryption, this should be given preference over that done directly by the Cloud provider (Cloud Security Alliance, 2012), since it ensures data confidentiality starts as soon as the data leaves the direct control of the Cloud consumer. For instance, this is the approach recommended to protect the sensitive data in the hybrid SaaS conceptual Model B. Securing data through client side encryption is necessary for compliance with certain legislation, such as HIPAA. However, client side encryption limits the operations that the Cloud provider can perform on the data, such as searching and indexing (refer to Section 5.1), making this type of encryption not always possible.

### 7.5 Legislation and Compliance

In the case of Cloud encryption, it is also recommended that encryption algorithms and keys used should be selected such that these ensure compliance with legislation across the different jurisdictions where the Cloud data might be stored or processed (European CIO Association, 2012). As discussed in some detail in Section 5.3, compliance in a Cloud environment might be difficult to achieve, so decisions related to encryption should be done with care. Expert legal advice should be sought, as well as recommendations from the Cloud provider, in order to at least meet the minimum requirements to achieve compliance.

This best practice is the least relevant in conceptual Model D, since this is a private infrastructure held completely on premise at the consumer. On the other hand, this is particularly applicable in public Clouds, where the data is moved into a shared Cloud infrastructure over which the consumer has limited control. Moreover, this best practice is especially relevant to cases similar to conceptual Model B, where a hybrid approach is purposely used due to the sensitivity and confidentiality of the data being processed.


**7.6 Encryption that Enables Easier Management**

Although the Cloud is a relatively new concept in computing, the building blocks used by this model are not new. When it comes to Cloud encryption, various techniques or types of encryption can be applied just as they would be used in traditional computing, possibly with some slight adaptations to ensure security even in the Cloud model. Apart from public key cryptography and symmetric key cryptography discussed in Section 7.2, the Cloud can leverage upon various other types of encryption. Two such examples are Identity Based Encryption and Attribute Based Encryption, which are described in Section 7.6.1 and Section 7.6.2, respectively.

### 7.6.1 Identity Based Encryption

Identity Based Encryption (IBE) is an approach to public key cryptography in which the public key consists of an arbitrary piece of information which identifies the user. Such examples which are often used in IBE include an email address or a telephone number (Shamir, 1984). By using this information as the public key, a public key infrastructure which does not rely on certificates can then be used, resulting in an environment which is more lightweight and easier to manage (Li, Dai, & Yang, 2009). Such an approach makes it, therefore, possible for third parties to encrypt messages without needing an underlying infrastructure to distribute and verify public keys.

Identity Based Cryptography was first proposed by Shamir, which made use of an email address as the public key (Shamir, 1984). This, however, only supported digital signatures and not encryption, and it was only several years later that solutions to IBE were proposed. The first approach performed encryption one bit at a time, something which was inefficient and resulted in significant ciphertext expansion (Cocks, 2001). Another approach was also proposed which was provably secure and relied on more efficient mathematical constructs (Boneh & Franklin, 2001).

In IBE, the public key is derived using some well-known information about the user as well as a public master key. This public master key is published by an entity known as the Private Key Generator (PKG), who is also responsible for generating the private keys. In fact, this PKG derives the private key from the public one and a secret private master key (Li, Dai, & Yang, 2009). This private key is then passed on to the legitimate user who owns the public identity. The PKG is, therefore, considered to be a major player in IBE and it is trusted to maintain the private master key and the private user keys secure and confidential. Moreover, the PKG needs to authenticate the user before issuing a private key for a public one, in order to ensure that a private key is not issued to an unauthorised entity.

An IBE algorithm typically consists of four main components (Boneh & Franklin, 2001):

- **Setup** – This task is used by the PKG in order to initialise the IBE architecture, including the master keys
- **Extract** – This algorithm is used to generate a user’s private key from the public identity key
- **Encrypt** – This converts a plaintext message into ciphertext, using the user’s public key
- **Decrypt** – This algorithm converts ciphertext back into plaintext, using the user’s private key

IBE, therefore, makes it easier to implement public key cryptography as no certificate management is needed, since publicly known identifiers are used. This approach to encryption is useful in situations where multiple actors are involved in the encryption and decryption process, such as in public or hybrid models. Cases similar to conceptual Model A and Model B, which are a
public and hybrid SaaS model respectively, can make effective use of IBE, since this provides all the benefits that public key cryptography brings whilst simplifying the public key management process. The limitation, however, is that mechanisms such as key revocation are not inherently part of IBE and alternative approaches need to be considered. Moreover, in IBE the PKG is not only trusted to maintain the private master key secure, but it also knows the user’s private key since it is responsible for deriving this for the user. A way to ensure the secure delivery of the private key from the PKG to the user also needs to be considered. Moreover, if the PKG is compromised, then all of the user private keys that were generated by it over time will also be compromised. Trust, therefore, plays a major role in IBE.

Since IBE is an interesting prospect for the Cloud, research in this area is an on-going process. For instance, a Hierarchical Identity Based Cryptography (HIBC) approach was proposed, which combines IBE with federated identity management to simplify key distribution (Yan, Rong, & Zhao, 2009). Another approach known as the Hierarchical Architecture for Cloud Computing (HACC) was also proposed, which creates a hierarchy of PKGs to facilitate the IBE process and ease bottlenecks (Li, Dai, & Yang, 2009). This approach makes use of secondary slave PKGs to manage different parts of the infrastructure, rather than having a single PKG responsible for doing all the work. These new approaches to IBE can, therefore, help reduce the disadvantages that IBE offers, making this a feasible approach for implementing encryption in a Cloud environment.

### 7.6.2 Attribute Based Encryption

Another approach to encryption that can be used in a Cloud environment is Attribute Based Encryption (ABE), which is a type of functional encryption. Functional encryption is an approach which makes it easier to share encrypted data, since it makes use of some “functionality \( f(x,y) \) which determines what a user with secret key \( y \) can learn from a ciphertext encrypted under \( x \). This allows an encryptor to specify a policy describing what users can learn from the ciphertext, without needing to know the identities of these users or requiring them to have already set up public keys” (Lewko, Okamoto, Sahai, Takashima, & Waters, 2010).

ABE, therefore, aims to address the problem associated with traditional public key encryption, which necessitates that data is encrypted with a specific user’s public key and then decrypted with its corresponding private key, making it difficult to define flexible access control policies related to the ciphertext (Ostrovsky, Sahai, & Waters, 2007). ABE, which was introduced by Sahai and Waters, addresses this situation by associating a number of attributes with the ciphertext (Sahai & Waters, 2005). Users are then given one or more private keys, with each one enabling access to data that has specific attributes. The data, therefore, can be decrypted only if the key satisfies possibly complex logic expressions defined over its attributes, using logic gates and threshold values (Ostrovsky, Sahai, & Waters, 2007). In this approach the encryptor can thus define a policy stating what type of users would be able to access the data (Lewko, Okamoto, Sahai, Takashima, & Waters, 2010).

Consider the example where a sensitive document of the secret service is to be encrypted. This document may have associated with it attributes such as “Top secret AND (Military OR All)”. If a user key has attributes “Military” and “Top secret”, then this can be used to decrypt this document. There is, however, a limitation in that the logic expressions cannot support negation in an easy and efficient way. This is because the absence of an attribute in the key does not mean that it this does not apply, but it merely means that the attribute was not considered to be relevant when the list of attributes was being drawn up (Ostrovsky, Sahai, & Waters, 2007).
Cloud Encryption and Key Management Considerations

Similar to IBE, the keys are managed by a central authority which is sufficiently trusted to generate and issue keys. This process relies on specific attributes and policies associated with each user. On the other hand, the encryption process can be performed by any user using a public key, assigning attributes to the ciphertext and defining a confidentiality policy in the process.

There are two main types of ABE, namely Ciphertext Policy Attribute Based Encryption (CP-ABE) and Key Policy Attribute Based Encryption (KP-ABE) (Goyal, Pandey, Sahai, & Waters, 2006). “In CP-ABE system, keys are associated with sets of attributes and ciphertexts are associated with access policies. In a KP-ABE system, the situation is reversed: keys are associated with access policies and ciphertexts are associated with sets of attributes” (Lewko, Okamoto, Sahai, Takashima, & Waters, 2010). Therefore, in CP-ABE, each user is assigned a number of attributes and the central key authority gives the user a number of keys that are based upon the user’s attributes. The ciphertext then has associated with it a set of attributes which form a policy, constructed from a logic expression on these attributes. If the user’s attributes satisfy the policy, then the key can be used to decrypt the ciphertext. However, multiple users or keys cannot be combined together to satisfy the security policy. On the other hand, in KP-ABE attributes are assigned to the ciphertext directly. The central key authority specifies the attributes that each key should have, together with the policy associated with each key. The limitation with KP-ABE is that the data owner doesn’t have complete control over the access policy, since this is left in the key authority’s hands (Darnasser, 2012). However, whilst these two types of ABE are effectively the opposite of each other, it is also possible to convert a KP-ABE system into one which is CP-ABE (Goyal, Pandey, Sahai, & Waters, 2006).

An ABE system typically consists of five main operations (Darnasser, 2012):

- **Setup** – This operation initialises the ABE system, generating a master key which is retained by the central key authority, as well as the public key used for encryption.
- **Encrypt** – This operation takes as input the public key, a plaintext message and an access policy, and it encrypts the message with the public key and associates the policy with it.
- **Key Generation** – This task is performed by the central key authority, and it generates a private key for a particular user, using the master key in the process. The key generation operation requires as input the attribute list of the user.
- **Decrypt** – This operation is the opposite of the encryption function, and it compares the private key’s attributes with those of the ciphertext.
- **Delegate** – This operation allows a user to generate private keys that have privileges which are less than or equal to those that the user already has, allowing the user to generate less privileged keys which can be delegated to other users.

A limitation with ABE is that even if a user cannot decrypt the data due to insufficient attributes, the user can still learn the logic expression associated with the ciphertext (Ostrovsky, Sahai, & Waters, 2007). This side channel results in the leakage of information about the possible contents of the encrypted data. Predicate Encryption (PE) is a variation of ABE which aims to keep this expression secret from users who cannot decrypt the ciphertext, keeping the ciphertext policy and attributes confidential. In PE, the private keys correspond to predicates which enable decryption to take place (Katz, Sahai, & Waters, 2008).

In a Cloud environment, ABE can help facilitate encryption and key management processes by associating access policies directly with the keys or with the ciphertext. An example where ABE can be used would be in conceptual Model C, which is a public PaaS system, where an organisation can use a file storage service to upload documents to the Cloud. If these documents
are sensitive, then ABE can be used to encrypt the data to ensure their confidentiality against unauthorised users. In the process, a security policy will be associated with these encrypted files, defining the attributes that are necessary to decrypt the files. The authorised users will then each be issued with private keys that enable different users to decrypt different files, as long as the policy is satisfied. Similarly, ABE might be useful for SaaS or IaaS models which need to support complex policies that cater for multiple users. This can not only be used in public or hybrid models, but also in private deployment models.

An extension to ABE known as Temporal Attribute Based Encryption (TABE) also allows for policies which are not only based upon logic Boolean functions on attributes, but also upon numeric comparisons. These policies, therefore, make it possible for particular files to be decrypted with specific keys only during predefined date or time ranges (Zhu, Hu, Ahn, Gong, & Chen, 2011). An example of such a policy would be that a particular file can only be decrypted by users with the “Engineer” attribute, from Monday to Friday between 0800hrs and 1700hrs.

ABE, therefore, is yet another active area of research in Cloud Computing, which can help simplify encryption and key management, by allowing decryption to take place based upon access policies that are inbuilt directly into the encrypted data.

### 7.7 Encryption Interoperability

Another factor that should be given priority when implementing Cloud encryption is the concept of interoperability. Since encrypted data eventually needs to be decrypted, it is considered best practice to use standard approaches to encryption, allowing the data to be decrypted by different entities such as the customer or a trusted third party, given that the decryption key is known (Cloud Security Alliance, 2012). Such an approach ensures that data can be decrypted without being dependent upon a single entity. In particular, therefore, this applies to public and hybrid infrastructures of all service models.

Interoperability of the encryption format and file structure used can be achieved using file formats such as OpenPGP or Secure ZIP (Cloud Security Alliance, 2012), providing the advantage that such files can be understood and processed by various applications and services in a standard way. Interoperability can also be achieved in the use of encryption algorithms, by applying standard encryption algorithms such as AES and RSA. In fact, when encryption is performed by the provider, this will often support various encryption formats, allowing the customer to choose the ideal mechanism for the particular situation. Moreover, key management should also follow a standard approach. Standard key formats such as X.509 and OpenPGP should ideally be used in Cloud encryption, together with interoperable key management processes such as the Key Management Interoperability Protocol (KMIP) by OASIS (OASIS, 2010), as discussed in Section 6.8.1.

### 7.8 Implementation of Desirable Features

When using Cloud encryption, a user expects that this does not limit the functionality of the system. For instance, if when the Cloud data in unencrypted, a user would normally be able to use the data directly on the Cloud without needing to first download it, then techniques should be sought that help achieve this even when encryption is used. Similarly, a user expects to be able to sort and search through the data, irrespective of whether or not this is encrypted. Although features such as these are not critical for encryption to operate successfully, as discussed in in Section 5.1 these are highly desirable yet challenging to implement. Despite this, it is considered to be a best practice to take all appropriate technical and procedural measures to ensure that
encryption is not a burden on the user and to address, as much as possible, the limitations that encryption inherently introduces.

Ways of implementing this desirable functionality are, however, still considered to be active areas of research in the domain of Cloud Computing. In fact, whilst established mechanisms exist to implement the essential best practices for Cloud encryption and key management previously described, techniques for adopting the desirable features of Cloud encryption are still in their infancy and often not yet sufficiently refined to support a complete and standardised implementation. This section, therefore, will introduce some areas of active research, which aim to make it possible to provide the desired functionality to the user in the near future.

7.8.1 Processing Encrypted Data

A major limitation of Cloud encryption is that the provider can only perform very limited meaningful processing on the encrypted data it stores, unless it knows the decryption key (refer to Section 5.1). Since it is not always desirable, or indeed possible, for the Cloud consumer to supply this key to the provider, alternative ways are being researched which allow the provider to process this encrypted data in a way that still maintains its confidentiality. Fully Homomorphic Encryption (FHE) is one such approach, which makes it possible to perform complex mathematical operations on encrypted data, without the need of having to decrypt this beforehand and without compromising the encrypted data’s security in the process (Brooks, 2009).

Gentry, who first proposed a scheme for FHE, justifies his research by stating that “an annoying property of encryption is that it typically traps data inside a box that prevents the data from being used or analysed until you open the box with the secret decryption key. What a fully homomorphic encryption scheme allows you to do is analyse or compute a function of the data while it remains securely inside the box” (Prince, 2009). An implementation for FHE would, for instance, be suitable in the medical sector where personal health equipment could be used to monitor a patient’s health conditions and upload them to the Cloud in near real time. Such data should be encrypted to safeguard the patient’s confidentiality, and should only be decipherable by the patient and authorised health care professionals. However, it would also be desirable for the Cloud to be able to trigger an alarm if unusual patterns are observed in the data, which could indicate a medical emergency. This can be achieved using FHE, which can identify any significant deviations from the average values of a particular patient, without needing to know what these average values are and therefore having to decrypt the enciphered data.

Gentry’s approach to FHE is based upon public key cryptography using the RSA algorithm. In fact, the concept of FHE (then known as privacy homomorphism) was first introduced by Rivest, Adleman and Dertouzos (Rivest, Adleman, & Dertouzos, 1978) after RSA was proposed. RSA, however, only supports multiplicative homomorphism. This means that it is possible to multiply two ciphertexts generated using the RSA algorithm in order to produce the ciphertext that would be obtained by encrypting the product of the two original plaintexts. Similar to ElGamal, which allows for the multiplication of encrypted messages, RSA is said to be a Partially Homomorphic Encryption scheme (Atayero & Feyisetan, 2011). FHE, on the other hand, allows both addition and multiplication to be performed on the ciphertext directly, using binary XOR and AND functions, respectively (Gentry, 2009). A FHE scheme is one which satisfies the property $E(m_1 \otimes m_2) = E(m_1) \otimes E(m_2)$, where $E(m)$ signifies the encryption of message $m$, and $\otimes$ represents an arbitrary function such as multiplication or addition (Atayero & Feyisetan, 2011).

In FHE, it is possible to perform mathematical operations directly on the encrypted data, since the plaintext and the ciphertext are said to be homomorphic, meaning that they are of the same
Cloud Encryption and Key Management Considerations

mathematical shape (Brooks, 2009). The complexity in such an approach lies in the fact that, however, this property should not render the encryption system less secure by leaking sensitive information about the ciphertext.

Gentry’s approach to FHE consists of four main operations (Gentry, 2009):

- **Key Generation** – This operation creates a key pair \((sk, pk)\), where \(sk\) is the secret private key and \(pk\) is the public key
- **Encrypt** – This operation encrypts a message using the secret key, in order to obtain a ciphertext
- **Decrypt** – This operation decrypts a ciphertext using the private key, in order to obtain the original plaintext message
- **Evaluate** – This operation generates the ciphertext that would be obtained by encrypting a message modified by some function, by looking only at the ciphertext obtained by encrypting the original message. FHE schemes require that this evaluation operation works on any arbitrary function, which in turn consists of a sequence of addition or multiplication operations.

Apart from being based upon the RSA algorithm, Gentry’s FHE scheme is also built upon lattice-based cryptography and ideal lattices (Gentry, 2009). In lattice-based cryptography, the encryption process results in the selection of a location next to a point on the lattice. Decryption, on the other hand, involves eliminating this noise by finding the actual point. Such cryptography schemes are secure since an attacker doesn’t know the structure of the underlying lattice, so it is difficult to find the lattice point without additional information such as the decryption key (Finney, 2009). The addition or multiplication of encrypted data effectively results in the selection of a new location on the lattice, with the resultant noise being the summation or product of the individual noise parameters, respectively (Gentry, 2009). If the resultant noise is too large, then it will be impossible to decrypt the ciphertext obtained, by finding the correct point on the lattice. FHE, therefore, works by attempting to reduce the noise of the individual components before computing the summation or product of these ciphertexts. This is performed by a fifth FHE operation known as Recrypt, which refreshes a ciphertext to produce another one using a different key but which has less noise (Atayero & Feyisetan, 2011). This process results in the FHE scheme being bootstrappable, meaning that it can evaluate the decryption algorithm (Atayero & Feyisetan, 2011).

Following Gentry’s scheme for FHE, a number of variations were put forward. Amongst others, these include a specialisation of the scheme which results in smaller ciphertexts (Smart & Vercauteren, 2010) and a scheme which works on integers instead of at the bit level (Van Dijk, Gentry, Halevi, & Vaikuntanatha, 2010). A Homomorphic Encryption Scheme using Residue Number System (HORNS) was also proposed as a variant of Gentry’s FHE, which leverages upon the smaller and parallelised arithmetic operations required in residue number system theory, in order to provide a performance improvement (Gomathisankaran, Tyagi, & Namuduri, 2011).

When considering the conceptual models discussed in this study, FHE would be most relevant for Model B which is a hybrid SaaS model. This is because the Cloud consumer could encrypt the sensitive data that is stored locally and then transmit this to the SaaS application at the provider so that it can be processed and used. Through FHE, the confidentiality of the data may be preserved since the provider will be able to perform some processing on this data without the need to first decrypt it. In exactly the same way, FHE can be applied to all data processed in Model A which is a public SaaS model. FHE can also be used in situations similar to Model C and Model E, which are public PaaS and IaaS models respectively, although FHE might limit the
meaningful processing that can be done when such low level encryption is used. In Model D, which is a private IaaS model, the complexity of FHE might outweigh the benefits provided, seeing that such a Cloud setup is completely under the direct responsibility of the Cloud consumer organisation.

Despite its clear desirability in a Cloud environment, whilst FHE is theoretically possible it currently is not sufficiently refined to enable actual implementations. This is because known FHE algorithms currently require excessively large amounts of computing processing power which even go beyond what a Cloud infrastructure can effectively handle in practice. However, active research is being carried out in this area in order to find more efficient ways to implement FHE, enabling the Cloud provider to process and use the encrypted data without being able to actually decipher it.

7.8.2 Searching and Sorting Encrypted Data

As discussed in Section 5.1, encryption of Cloud data introduces a number of limitations, most notably the inability to access data through search. This is a major disadvantage of Cloud encryption, since users often tend to locate data by searching for it. In order to work around this limitation, an index containing the search keywords can be stored locally by the consumer, but this does not leverage on the benefits of the Cloud. For this reason, approaches to Cloud encryption which allow searching and sorting to be performed directly by the provider are being researched. One such approach allows the users to create an index upon their encrypted data, and then upload this index and the encrypted data to the Cloud. The user can then generate credentials and tokens that are used to determine who can access the data, and what encrypted data results will be returned for particular keyword searches.

Searchable encryption schemes make use of an encrypted index which is kept confidential by the Cloud provider, and which cannot be read unless an appropriate token is provided (Kamara & Lauter, 2010). Only by providing a token does a user enable the provider to find the pointers to the encrypted files where the keyword is found. The generation of these tokens, therefore, needs to be controlled since these are considered to be the trapdoors that enable the encrypted index to be used. The generation of the tokens relies upon a secret key.

A Cloud encryption architecture which makes search possible, typically consists of the following main components (Kamara & Lauter, 2010):

- **Data Processor** – Manipulates the data before this is transferred to the Cloud
- **Data Verifier** – Verifies that the encrypted data on the Cloud has not been modified
- **Token Generator** – Constructs tokens that allow the Cloud provider to search the data
- **Credential Generator** – Generates credentials that enable decryption

As discussed in (Kamara & Lauter, 2010), in such an environment, the Cloud consumer first generates a master key and this is stored locally. The data is encrypted by the Data Processor using the key and a suitable encryption algorithm such as AES, before uploading it to the Cloud. Metadata, such as search keywords stored in an index, are also uploaded to the Cloud, encrypted with a searchable encryption scheme. Once on the Cloud, the user can then search for this data. The search term and the key are both passed through the Token Generator, which generates a token that enables searching to take place. This is sent to the Cloud provider who uses it to return a list of encrypted files that contain the search term, by using the metadata associated with the files. The Credential Generator is then invoked to derive credentials from the master key using attribute-based encryption, in order to allow for the encrypted files to be decrypted. This
Cloud Encryption and Key Management Considerations

makes it possible to generate credentials for another user, without needing to disclose the master key. Using such a high level approach, it is therefore possible to have encrypted data stored on the Cloud, which the provider can search in without needing to decrypt the file’s contents. Search upon encrypted files becomes a possibility when the files are encrypted in such a way that enables search, and when the search term is processed before using it to look through the metadata.

The two types of searchable encryption schemes are Asymmetric Searchable Encryption and Symmetric Searchable Encryption (Kamara & Lauter, 2010).

Asymmetric Searchable Encryption (ASE)
In this type of encryption, the users encrypting and decrypting the data may be different, so public key cryptography is used. The encrypted search terms that act as the trapdoor to process the encrypted index, can be generated by someone who has access to the private key. However, knowledge of the public key is sufficient for anyone to add words to the search index (Curtmola, Garay, Kamara, & Ostrovsky, 2006). When compared to Symmetric Searchable Encryption, ASE supports more functionality, although it can be less efficient when it comes to encryption due to the inefficiencies associated with public key cryptography (Kamara & Lauter, 2010). Efficient ASE (ESE) schemes exist, but these are suitable only when the keywords are complex and difficult to guess.

Symmetric Searchable Encryption (SSE)
In this encryption scheme, symmetric cryptography is used. Since the same key is used for encryption and decryption, this technique is usually used when the data is searched for and decrypted by the same user who encrypted it in the first place. This means that data can be structured in any arbitrary way before being decrypted, such that storage and searching are as efficient as possible for the particular scenario (Curtmola, Garay, Kamara, & Ostrovsky, 2006). SSE is based upon block ciphers and pseudo random functions, which result in a scheme which is efficient when it comes to encryption (Kamara & Lauter, 2010).

In SSE schemes, five basic operations are involved (Curtmola, Garay, Kamara, & Ostrovsky, 2006):

- **Key Generation** – This involves using probabilistic key generation techniques upon a master key, in order to generate a secret key $K$ which will then passed as input to the other operations
- **Encryption** – This uses a probabilistic algorithm to encrypt a number of files using the key $K$. This operation gives as output the encrypted files, as well as the secure index. The secure index is an encrypted index which only allows search operations to be conducted on it, if the trapdoor is used.
- **Trapdoor Generation** – This is a deterministic algorithm which generates a trapdoor based upon a keyword and the key $K$
- **Search** – This involves a deterministic algorithm that takes as input a trapdoor and a secure index, and returns a list of files containing the search term.
- **Decryption** – This is a deterministic algorithm that decrypts a retrieved document using the key $K$.

CS2 is an implementation of a searchable cryptographic Cloud storage system (Kamara, Papamanthou, & Roeder, 2011). This algorithm allows for data to be stored on the Cloud in a way that ensures confidentiality and integrity, and which also allows for data to be verified to ensure that the CS2 operations were performed correctly by the Cloud provider. Moreover, the
algorithm allows for encrypted data to be searched based upon keywords, and for the associated index to be updated easily and dynamically (Kamara, Papamanthou, & Roeder, 2011).

A limitation of searchable Cloud encryption in general, is that this performs the matching process by comparing the search term with the keywords stored in the index. Fuzzy searching is desirable since it enables the search process to handle situations where the search term is slightly misspelt or not exactly the same like the keywords stored in the index (Li, et al., 2010). For such a fuzzy search implementation to work, one approach would be to modify the index and for each existing keyword, add all possible words that derive from it and which have an edit distance of not more than an arbitrary value. This, however, is inefficient since the index will become too large to maintain, difficult to manage and slow to use. Assuming a distance of 1 the keyword CLOUD will result in words such as ALOUD, BLOUD, CLOUD, DLOUD, and so on. Instead, wildcards can be used to make this more efficient. For instance, the keyword CLOUD will result in the variations CLOUD, *CLOUD, *LOUD, C*LOUD, CL*UD, CLO*D, CLOU*, and CLOUD*. When a fuzzy search is being performed on the Cloud, all variations of the search term using wildcards are generated, and the trapdoor for each is computed. These are then passed on to the search operation that applies the trapdoors to the keywords index to identify which files contain one or more of the search term variations (Li, et al., 2010). While performing the matching process, the search process knows how to handle and match these wildcard characters, and also sorts the returned results according to the edit distance in descending order.

Searchable encryption schemes can be used in all types of service and deployment models. In particular, SSE schemes are particularly relevant to situations similar to conceptual Model D, which is a private IaaS infrastructure. This is because in this model, data is uploaded and retrieved by the same actor, meaning that there are no complex key sharing processes to consider. Similarly, Model A can also benefit from SSE since encryption and searching are both done by the Cloud provider. On the other hand, ASE schemes are particularly useful in models where encryption is done by the Cloud consumer and search is processed by the provider. The hybrid SaaS conceptual Model B is one such case, where the provider does not need to know a shared symmetric key to be able to process the search, but instead makes use of public key cryptography.

Searchable encryption schemes, therefore, make the prospect of Cloud encryption more appealing to the consumer, since they provide all the benefits of Cloud encryption together with the added convenience that search functionality provides. Various types of searchable encryptions schemes exist, including symmetric and asymmetric encryption. Techniques such as fuzzy matching can also be used to enhance the functionality of such searchable encryption schemes, encouraging further the use of Cloud encryption.

7.9 Conclusion

When implementing Cloud encryption, appropriate measures should be taken to ensure that this is done in way which limits issues related to information security and usability. Various established best practices exist for traditional encryption systems, and most of these apply even in the Cloud. Other best practices specific to Cloud Computing, however, should also be considered. The establishment of an encryption policy is an essential best practice, since this defines and guides the encryption processes, which might make use of different types of encryption depending upon the scenario in question. As a best practice, the integrity of encrypted data should also be preserved to ensure that this has not been tampered with. Client side security should also be considered, since failure to do so can give rise to confidentiality,
integrity and availability threats. Best practices also state that all appropriate measures should be taken to ensure compliance with existing legislation.

Since Cloud encryption is necessary yet challenging to implement, any techniques that simplify its implementation and management should also be considered. Such examples include Identity Based Encryption and Attribute Based Encryption, which allow for simplified key management and encryption policies, respectively. Interoperability of encryption processes used should also be given its importance, as should approaches which introduce additional functionality to prevent encryption from being too limiting for the end user. Such techniques include searchable encryption techniques and Fully Homomorphic Encryption, which provides the ability to process encrypted data directly. Whilst to date, such processes are often not sufficiently refined for a Cloud environment, it should be considered best practice to regularly analyse and assess such technological offerings, in order to ensure that the best possible encryption experience is provided to all actors involved in the Cloud.
8 Dropbox – A Cloud Encryption Case Study

Dropbox is a Cloud-based file hosting service that allows users to automatically synchronise the contents of a local folder with a repository on the Cloud. This product is considered to be a successful Cloud application, since its user base has grown to over one billion users since its inception in 2008 (Schramm, 2012).

As shown in Figure 21, a Dropbox user can place a file in a local folder and this will get uploaded to the Cloud through the use of an application installed on the client. The upload process is transparent to the user, as long as the account has sufficient unused Cloud storage space available, irrespective of whether it is part of the free or paid plans offered by Dropbox. Moreover, a user can also manually upload files to the Dropbox Cloud through the product’s website. If the same Dropbox account and client software are setup on multiple computers, then these will synchronise their contents with the Cloud account, until all computers as well as the Cloud have the latest version of the folder’s contents (Dropbox, 2012). Once synchronised, the user can then access the files directly from the local folder. Moreover, the files can also be fetched from the Cloud through the Dropbox website or using a mobile device that has a lightweight version of the Dropbox client installed.

In this Chapter, an analysis is done of Dropbox’s data confidentiality measures, using the approaches described in this study. Section 8.1 classifies Dropbox using the taxonomy provided in Section 2.5, whilst a CC-RAM analysis of Dropbox is done in Section 8.2, during which references are made to the threats and best practices related to Cloud encryption, as discussed in this study. Section 8.3 concludes this Chapter by providing a short overview of the security measures taken by some competitor Cloud storage products.

8.1 Cloud Classification

Dropbox is a Cloud application that can be accessed through a web browser. The user does not need to consider or manage any technical aspects of the system in order to be able to use it, but rather simply registers for the service and starts uploading files to the Cloud application. For these reasons, Dropbox is classified as a product that falls under the SaaS Cloud service model, as
described in Section 2.3. More precisely, according to the taxonomy provided in Figure 2 of this study, Dropbox falls under the *Backups* category of the SaaS service model.

For added convenience the service need not be accessed only through a web browser, but Dropbox also provides the option of a non-mandatory client application that can be used to facilitate and automate Cloud communication and file synchronisation processes for the user.

### 8.2 CC-RAM Analysis of Dropbox

Following an initial, high level analysis of Dropbox, it becomes evident that a more detailed risk assessment is required since various security risks might be involved. This can be conducted using the Cloud Computing Risk Assessment Method (CC-RAM), as described in Section 3.4.

#### 8.2.1 Asset Identification

In the first phase of CC-RAM, the main assets involved in Dropbox are identified, in order to set the scope and boundaries of the risk assessment process. These assets can be broadly categorised into data and processes.

The data assets in Dropbox include the files that are stored on the local machine as well as the copy of the files that reside in the Cloud. Moreover, the users’ account credentials are also data assets that need to be kept secure. The client software that is installed on the user’s machine as well as the Dropbox website that is used to manage the files and user account are two more data assets whose security needs to be considered.

Apart from being data, assets can also be processes. In Dropbox, the user authentication process is an important asset which is necessary for any interaction with the Cloud to be allowed. The file upload, download and synchronisation processes are the main essential business operations of the Dropbox system. The file sharing process is yet another asset which is provided by the system, which allows the user to share his files with one or more third parties.

#### 8.2.2 Asset Valuation

The assets previously identified need to be valued according to their level of importance to the organisation. As discussed in Section 3.4, this valuation process can be done at a high level or in more detail using a valuation methodology, especially if the organisation has such methodologies already in place.

In this case study, it is assumed that the data being placed in Dropbox is of high business value and of a confidential nature. On the other hand the automatic file synchronisation client software is considered to be of medium to high value to the organisation, since it provides the required convenience for the user. With regards to process assets, the file upload, download and synchronisation processes are all considered to be of high value since they provide the core functionality of the system, while file sharing is of lesser importance. File access using the Dropbox website is considered to be of low to medium value, since it is not something that the organisation intends to use frequently.

The user’s data files, therefore, are the main asset in this Dropbox scenario, and this should guide the rest of the risk assessment process. On the other hand, the organisation might be willing to take more risks with other data or process assets that are less valued, as long as these calculated risks do not compromise the security of the main data asset in any way.
8.2.3 Deployment Model Analysis

Dropbox makes use of a public Cloud deployment model. This not only means that the Cloud is shared across multiple tenants, but also that a copy of the data is stored off premise. As discussed in Section 2.4, this model is more flexible and scalable than its private counterpart, and it allows for greater economies of scale to be achieved. However, this deployment model is more complex to manage securely seeing that the infrastructure is shared with multiple users, some of which might be malicious. The Cloud consumer also has less control over how the Cloud is managed, and where on the Cloud the data is stored or processed. Dropbox does not support private Clouds (Dropbox, 2012).

8.2.4 Service Model Analysis

As discussed in Section 2.3, SaaS, PaaS and IaaS provide the different levels of user control on the data and the Cloud infrastructure. Dropbox makes use of the SaaS service model, which gives the user the least Cloud control. In this application, data security is mainly the responsibility of the Cloud provider, since this manages the data once it is uploaded to the Cloud.

8.2.5 Data Flow Identification

During CC-RAM, the data flows involved need to be identified so that they can be evaluated and secured. In Dropbox, these flows include file upload to the Cloud as well as file download, both of which can be done manually or through the automatic synchronisation process. Another external data flow is file sharing, which results in other users gaining access to one or more files stored on the Cloud. As discussed in Section 3.4, internal data flows should also be analysed in CC-RAM. In Dropbox, these data flows include the transfer of data from the web frontend servers to the Amazon S3 storage servers, and vice versa. Data can also flow internally within the provider’s infrastructure, such as to a file analysis or indexing server to make storage and retrieval faster and more efficient. Moreover, the user data might flow to the provider’s employees, when these need access to the data for administration, support or legal reasons.

8.2.6 Threat and Vulnerability Analysis

In this phase of CC-RAM, the threats and vulnerabilities that affect the previously identified assets and data flows in the Cloud service and deployment model used by Dropbox, are identified. The Cloud security concerns that are discussed in Section 3.2 and which are applicable to Dropbox security, are revisited in this Section.

Note that threat and vulnerability analysis would typically cover all security aspects of the product, although in this study, only the areas where encryption is of relevance are reviewed.

Unknown Risk Profile

Once data is placed in Dropbox, then the data’s security is managed by the Cloud provider. This is a potential threat, since there is an unknown risk profile involved. As a result, an in-depth investigation into the security measures adopted by Dropbox should be undertaken, in order to better understand the risk profile involved. This assessment should include a review of the technology, solutions or procedures used to provide data security.
Data Security
Dropbox is based upon Amazon S3 storage, providing a multi-tenant application which allows multiple users to share the same Cloud infrastructure. Although each user should potentially only be able to access his files unless these have explicitly been shared, there is a concern that multi-tenancy might compromise the data’s security. By placing data on the Dropbox Cloud, this is subjected to significantly more risks, including deletion, modification and loss of confidentiality.

Access to Cloud Data
By default, a Dropbox account is accessed via a username and password. If these Cloud credentials are obtained by an attacker, then the attacker might access the user’s data. Password authentication is considered to be a weak authentication mechanism, since the authentication secret is disclosed in the process, making it possible for an attacker to obtain this by eavesdropping on the message communication. Secure channels for communication should, therefore, be in place.

Cloud Provider Insiders
In Dropbox, the user uploads a copy of the files to a repository on the Cloud. This makes it possible for employees at the Cloud provider to potentially gain access to the data, be it for legitimate or malicious purposes. In fact, Dropbox acknowledges the fact by stating that its “employees may access file metadata (e.g. file names and locations) when they have a legitimate reason, like providing technical support. Like most online services, [Dropbox has] a small number of employees who must be able to access user data for the reasons stated in the privacy policy” (Dropbox, 2012).

Forensics
Auditing of actions done on the Cloud is desirable, but not always easily achievable or possible in a Cloud environment. In Dropbox, if consumer and provider actions done on the data or the Cloud infrastructure setup are not audited, then there is a threat that the data’s confidentiality and integrity might be compromised and this might go unnoticed.

Shared Cloud Infrastructure
In Dropbox, the same Cloud infrastructure is shared across multiple users. Although the number of operations that a user can perform on Dropbox is limited to upload, download, search and file sharing, it is possible for the shared nature of the Cloud infrastructure to still give rise to security threats. If a malicious user manages to find a vulnerability or flaw in the client software or Dropbox web interface, then this might be used to potentially gain access to the core infrastructure and to other users’ files.

Cloud Management
The security of Cloud management messages also needs to be considered, including threats to their confidentiality, integrity, availability and authentication. In Dropbox these messages include user account provisioning and deprovisioning, as well as the management of account-specific settings, such as the pricing plan that is being used.

Non-Compliance
When using Dropbox, data is transferred and stored on servers located in the United States (Dropbox, 2012). This might be a security concern, especially if the data is confidential or personal as its security might be mandated by law. Therefore, as discussed in Section 5.3, when using Dropbox, noncompliance might be a possible threat unless appropriate measures are taken.
8.2.7 Risk Score Calculation and Risk Treatment Planning

The CC-RAM approach requires that a risk score is calculated for each of the threats previously identified. This score is based upon the impact that the threat could have on the organisation, as well as on the probability of the risk actually materialising.

The expected probability and impact that each threat has is shown in the table below, which rates each of these values as low, moderate or high. Low probabilities are those which are unlikely to take place, while high probabilities are those which are likely to occur. Moderate values lie in between. Similarly, when discussing impact, a low value means that this threat might result in some disruption, but which can be easily handled. A moderate impact is one which requires significant amounts of time and resources in order to recover from it, while a high impact threat is one which severely damages the business operations either financially or in terms of reputation.

<table>
<thead>
<tr>
<th>Threat</th>
<th>Probability</th>
<th>Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unknown Risk Profile</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>Data Security</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>Access to Cloud Data</td>
<td>Moderate</td>
<td>High</td>
</tr>
<tr>
<td>Cloud Provider Insiders</td>
<td>Moderate</td>
<td>Moderate</td>
</tr>
<tr>
<td>Forensics</td>
<td>Low</td>
<td>Moderate</td>
</tr>
<tr>
<td>Shared Cloud Infrastructure</td>
<td>Moderate</td>
<td>Moderate</td>
</tr>
<tr>
<td>Cloud Management</td>
<td>Moderate</td>
<td>Low</td>
</tr>
<tr>
<td>Non-Compliance</td>
<td>High</td>
<td>High</td>
</tr>
</tbody>
</table>

The most likely threats to take place are the unknown risk profile issues, data security issues and also possible non-compliance with legislation. On the other hand, unauthorised access to Cloud data by external attackers or insiders is less likely to occur. Similarly, those threats which require an attacker to use complex techniques in order to exploit flaws in the shared Cloud infrastructure to access the Cloud user’s data or account management settings, also have a moderate probability of occurrence. The lack of any forensic or auditing information in the case of a major issue is less likely to occur when using Dropbox.

On the other hand, the threats which have the highest impact on the organisation include the unknown risk profile as well as possible lack of data security. Access to the Cloud data by unauthorised users, as well as possible non-compliance might also have a large impact on the organisation, resulting in financial losses as well as poor reputation if confidential data gets disclosed to external users. The impact might, however, be less if the confidential data is accessed by Cloud provider insiders. Lack of sufficient forensic information following an attack might have a moderate impact on the organisation, since this might result in the inability to monitor data access or prosecute an attacker. An attacker that exploits a vulnerability in the Dropbox application or website might have a moderate impact, since there is a chance that the confidentiality of some data might be compromised. On the other hand, if the attacker modifies the user’s Cloud management information, the impact will probably be low as long as the business data itself is not accessed.

After assigning the probability and impact ratings, each risk is then rated using the risk matrix shown in Figure 22. The risk rating can be low, moderate or high. High risks should be avoided where possible by modifying the approach, or mitigated using appropriate countermeasures. Moderate and low severity risks should usually also be mitigated, but depending on the organisation’s risk appetite, the organisation might accept to take some risks.
When applying the risk matrix to each of the confidentiality threats faced when using Dropbox, it emerges that there are three high level risks, three moderate ones, and two low level risks, as shown hereunder. In this particular scenario, it is assumed that the risk treatment plan is to mitigate all high and moderate risks, and accept low level ones.

### throttle

<table>
<thead>
<tr>
<th>Threat</th>
<th>Risk Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unknown Risk Profile</td>
<td>High</td>
</tr>
<tr>
<td>Data Security</td>
<td>High</td>
</tr>
<tr>
<td>Access to Cloud Data</td>
<td>Moderate</td>
</tr>
<tr>
<td>Cloud Provider Insiders</td>
<td>Moderate</td>
</tr>
<tr>
<td>Forensics</td>
<td>Low</td>
</tr>
<tr>
<td>Shared Cloud Infrastructure</td>
<td>Moderate</td>
</tr>
<tr>
<td>Cloud Management</td>
<td>Low</td>
</tr>
<tr>
<td>Non-Compliance</td>
<td>High</td>
</tr>
</tbody>
</table>

### 8.2.8 Countermeasure Identification and Gap Analysis

After assigning a risk level to each of the threats faced by Dropbox and deciding upon the risk treatment plan, suitable counter measures need to be identified in order to mitigate these threats. As described in Section 8.2.7, only high and medium level risks will be addressed in this risk assessment. Dropbox currently takes various measures to address potential Cloud issues and, therefore, a gap analysis is required to identify any areas where Dropbox security falls short. During the gap analysis, best practices described in Chapter 6 and Chapter 7 and which apply to Dropbox, will be considered.

### Unknown Risk Profile

When moving to the Cloud, the consumer needs to trust the Cloud provider to implement appropriate and effective security mechanisms. Dropbox addresses this risk by assuring its customers that it adopts “the best tools and engineering practices available to build and maintain Dropbox” and that it employs “a number of physical, technical, and heuristic security measures to protect user information from unauthorized access” (Dropbox, 2012). Dropbox also has a dedicated security team to ensure that risks are identified and addressed. It publishes details of some of these technical and procedural counter measures, in order to encourage the adoption of its services even by security-conscious users. The level of trust that the consumer has in Dropbox is the most important consideration when deciding whether or not to trust this provider with the data. Dropbox, in fact, tries to leverage on its reputation and that of Amazon Web Services upon which it is built, in order to build this trust relationship. However, customers who are not comfortable with the unknown risk profile involved can opt to take security a step further by implementing client side data confidentiality measures. A gap analysis, therefore, indicates that whilst Dropbox attempts to address the unknown risk profile threat by informing its customers about the security measures taken, it does not directly provide ways to allow the user to take additional security measures, such as client side encryption before uploading the data to the Cloud. Third party products can, however, be used to achieve this.
Data Security
In order to address the data security threats, Dropbox technology should be based upon a reliable and robust Cloud platform. This product, in fact, is built upon Amazon S3 storage, which provides a strong security policy that Dropbox technology builds upon (Dropbox, 2012). The user’s files should be backed up in order to ensure availability, and this measure is implemented by Dropbox using the underlying features of Amazon S3 (Dropbox, 2012) (Amazon, 2008). When considering data confidentiality, measures must be taken to ensure that the files placed on the Cloud are accessible only by those who are authorised to do so. This measure is implemented by Dropbox which guarantees that other users cannot see the Dropbox files unless file sharing is explicitly set up (Dropbox, 2012). Encryption is one such data confidentiality measure that is needed to address this threat, and this should be governed by a policy that is applicable to data that is at rest and in transit (refer to Section 7.1). Encryption of data that is in use is not strictly necessary in Dropbox, since the data isn’t used directly on the Cloud but is instead first downloaded onto the client machine. One of the measures to ensure data confidentiality is to make use of standard, established and interoperable encryption algorithms (refer to Section 7.2 and Section 7.7). Dropbox uses “modern encryption methods to both transfer and store the data, [such as] Secure Sockets Layer (SSL) and AES-256 bit encryption” (Dropbox, 2012). This encryption, however, is done server side using a generic Dropbox encryption key (Borgmann, et al., 2012). In fact, Dropbox does not allow users to specify their own encryption keys (Dropbox, 2012). Whilst this approach makes searching and sorting of data as well as key management easier, this lack of separation of duties (as discussed in Section 6.7) is a gap in Dropbox’s security. This is because the consumer is not involved in the key management process, and is required to trust the provider to perform both encryption and key management duties.

However, the fact that Dropbox is responsible for key management makes this potentially simpler to secure since there are no other parties involved in this process. Dropbox should implement a key management policy, as discussed in Section 6.6, in order to ensure that the encryption keys used are maintained as secure as possible. These keys should be securely generated using true random number generators, and stored using hardware security modules to avoid the keys from being compromised (refer to Section 6.2). Key availability should be guaranteed using key escrow or having off-site secure backups, to ensure that the data can always be decrypted (refer to Section 6.3). Keys should be regularly changed to ensure that these are not compromised through prolonged use (refer to Section 6.5).

Access to Cloud Data
In order to address the threat of having an attacker gaining unauthorised access to a user’s Dropbox account, secure login mechanisms must be in place. Dropbox states that the “login is protected by many layers of security, including password and two-step verification” (Dropbox, 2012). Two-step authentication makes it considerably more difficult for an attacker to gain access to a user’s account, since it does not only rely on some information but also requires physical possession of an object such as a mobile phone SIM card. When enabled, therefore, this is an appropriate measure that secures the login process. However, the password itself is only covered by a weak password policy that must be reinforced to protect against unauthorised access, especially when two-step verification is not enabled (Borgmann, et al., 2012). Both the registration screens and the actual login mechanism make use of TLS, which gives a secure communication channel that provides confidentiality, integrity and non-repudiation (Borgmann, et al., 2012). In addition, Dropbox temporarily locks a user account after many failed logins, in order to protect against brute force password attacks.
Cloud Provider Insiders
Dropbox’s policy states that it restricts employees from accessing user data stored on the Cloud, unless this is necessary for technical support reasons (Dropbox, 2012). Whilst this constraint helps reduce potential insider attacks, it is not a complete restriction and therefore the threat still applies. Countermeasures that should be implemented to mitigate this threat include employee screening and training, as well as the gathering of forensic and auditing information on data operations done by the Cloud provider employees (refer to Section 6.9). Although Dropbox does not publicise these controls, it is assumed that Dropbox already carries out these tasks in order to safeguard its reputation.

Shared Cloud Infrastructure
Since the Dropbox Cloud infrastructure is shared across multiple users, it is possible for a malicious user to find and exploit a vulnerability or flaw in the client software or Dropbox web interface, to potentially gain access to the core infrastructure and to other users’ files. A countermeasure to address this threat is the hardening of the website and client application, in order to strengthen security and reduce vulnerabilities (refer to Section 7.4). This measure is already adopted by Dropbox, which regularly reviews and updates its website and application security (Dropbox, 2012).

Non-Compliance
If sensitive data that must be protected by law is placed on the Cloud, then there is a risk of non-compliance with such legislation. All data uploaded to Dropbox is stored on the Amazon S3 servers located in the United States (Borgmann, et al., 2012), and this can further complicate compliance if the data owner is from outside the United States (for example, from within the European Union). The ideal countermeasure is to avoid placing critical or highly sensitive data on Dropbox, unless this is protected using client side encryption prior to uploading it to the Cloud, as discussed in Section 7.5.

Various countermeasures, therefore, should be taken to address the security threats faced when using Dropbox. Whilst Dropbox already implements some of these measures, the gap analysis done in this Section shows that there are still a number of measures which are either not implemented or which are not sufficient to provide the required levels of security. Dropbox should seek to implement these security controls to provide a more secure and flexible service. On the other hand, the consumer should be aware of these limitations and take additional security measures where possible to mitigate these threats. Such measures might involve encrypting the data before placing it in Dropbox, or else not uploading sensitive data to the Cloud.

8.2.9 Feedback Loops
Dropbox technology is constantly evolving, with new features or technical changes being rolled out at times. Such changes might alter the risk profile of Dropbox, with this product becoming more secure in certain aspects, and potentially less secure in others. The feedback loop of CC-RAM should be used to constantly monitor the suitability of the security controls in place, and to take any necessary action to ensure the data’s security is maintained even after changes are done to Dropbox, the Cloud environment, or the consumer’s security requirements. This feedback loop involves restarting the risk assessment process, identifying the threats and defining the necessary countermeasures, after which a gap analysis needs to be performed again to compare the currently implemented controls with the required ones. Such a feedback loop should be a regular process, the frequency of which is dictated by the organisation’s security policy.
8.3 Security of Dropbox Competitors

CloudMe
CloudMe is a Cloud file storage service that allows for file upload and synchronisation using a client application similar to Dropbox. This product also supports the sharing of specific files or folders uploaded to the Cloud. CloudMe does not encrypt data as it is transferred from the client to the Cloud and vice versa, and it does not encrypt any files uploaded to the Cloud either (Borgmann, et al., 2012). This makes CloudMe significantly more vulnerable to data confidentiality attacks than Dropbox. All data is stored in a European data centre, in Sweden.

CrashPlan
CrashPlan is a Cloud storage service that aims to store file backups from a client computer onto the Cloud or another computer accessible via the internet. CrashPlan encrypts communication between the client and the Cloud, but makes use of a proprietary algorithm instead of standard approaches such as SSL/TLS (Borgmann, et al., 2012). This is a security risk since proprietary algorithms might be insecure or not sufficiently analysed. Unlike Dropbox, however, CrashPlan allows the user to specify which 448 bit encryption key should be used to protect the data stored on the Cloud (CrashPlan, 2013). By default, the account password is used as the encryption key, meaning that the provider can decrypt and access the data since the user’s account password is known to CrashPlan. However, the user can alternatively specify a private password which is unknown to the provider, increasing security at the expense of reduced convenience and less data processing functionality that CrashPlan can support. Similar to Dropbox, user data stored on the CashPlan Cloud is kept in a data centre located in the United States.

Mozy
Similar to CrashPlan, Mozy is another Cloud backup service which enables manual or scheduled updates of data to the Cloud. Like Dropbox, Mozy makes use of SSL/TLS to protect the communication between the Cloud and the client application (Borgmann, et al., 2012). Mozy supports two encryption algorithms, namely Blowfish and AES. When Blowfish is selected, a 448 bit encryption key selected by the provider is used, while when the user opts for AES encryption, a user-defined 256 bit key is used (Mozy, 2012). This provides more flexibility to the Cloud user who can, therefore, either opt to entrust the provider with the key management function or to manage this himself. If a user defined key is used, as with Dropbox, the data must first be downloaded to the client before it can be used. On the other hand, if the Cloud provider’s key is used, then this data can be used directly on the Cloud since the provider is capable of decrypting the data. The data itself is stored on servers within Europe and the United States.

TeamDrive
TeamDrive is a Cloud service which supports file storage, synchronisation and sharing. Like CrashPlan, TeamDrive uses a proprietary algorithm to secure communication between the client application and the Cloud, and this goes against encryption best practices. Similar to Dropbox, 256 bit AES encryption is used to protect the data stored on the Cloud (Borgmann, et al., 2012). However, instead of using one key for all customers, TeamDrive uses a different AES key for each user (TeamDrive, 2013). All Cloud data used by TeamDrive is maintained within data centres in the European Union.

8.4 Conclusion

When an organisation decides to place its data on the Cloud, a comprehensive risk assessment needs to be done in order to identify the risks involved and mitigate these using appropriate controls. In this Chapter, the Cloud Computing Risk Assessment Method has been applied to
Cloud Encryption and Key Management Considerations

Dropbox, which is used as a case study. The various data and process assets involved in Dropbox are first identified and valued, after which the deployment and service models used are identified. The data flows between the Cloud provider and the consumer, as well as internally within the Cloud infrastructure are then identified. The threats and vulnerabilities discussed throughout the study are then reviewed, and those applicable to data confidentiality in Dropbox are identified. The most probable risks in Dropbox are those related to the unknown risk profile, data security and non-compliance. These risks also rank as having the highest impact, together with possible access to Cloud data by malicious users. This information is then used to calculate the risk level for each risk in Dropbox, with various high and moderate risks being identified. Countermeasures are, hence, needed to mitigate these risks, and various recommendations done earlier in his study can help provide the required security. A gap analysis comparison of the measures with the security provided by Dropbox shows that a number of issues still exist, since some measures are either not currently implemented or not sufficiently refined. Further security measures, therefore, should be taken by the Cloud consumer when placing sensitive or confidential data on Dropbox. Despite this, a review of competitor products notes that even these suffer from similar security issues.

The concepts discussed throughout the whole study can, therefore, guide the risk assessment process in order to make an informed evaluation of confidentiality in Cloud offerings such as Dropbox, and to understand how encryption can be securely implemented such that it can meet its goals as a suitable security measure.
9 Conclusion

In this study, the concept of Cloud security was discussed, with particular emphasis done on the use of encryption and related key management techniques in order to address these issues. This analysis was directed by the three primary objectives that were set out at the beginning of this study.

9.1 First Objective

To define Cloud Computing, and to understand the security concerns involved.

The term “Cloud” has been widely used throughout this study, and the definition adopted in this research is that given by the Cloud Security Alliance, which defines the Cloud as “an evolving term that describes the development of many existing technologies and approaches to computing into something different. Cloud separates application and information resources from the underlying infrastructure, and the mechanisms used to deliver them”. Moreover, it “enhances collaboration, agility, scaling, and availability, and provides the potential for cost reduction through optimized and efficient computing... Cloud describes the use of a collection of services, applications, information, and infrastructure comprised of pools of compute, network, information, and storage resources” that can be acquired or released as necessary (Cloud Security Alliance, 2009).

Based upon this definition, six main Cloud characteristics were discussed (NIST, 2009) (Cloud Security Alliance, 2009). These are:

- **On-Demand Self-Service** – The ability for the consumer to request and consume resources whenever these are needed
- **Broad Network Access** – The Cloud is accessible over a network, and from various types of hardware or software
- **Resource Pooling** – Cloud resources are shared by several services, and they are allocated as necessary to meet the various demands
- **Rapid Elasticity** – A Cloud application can expand and contract its demand for required resources
- **Measured Service** – Resources used are metered and the consumer is charged accordingly
- **Multi Tenancy** – A single Cloud service instance can be used to serve several clients, with the different tenants being isolated from each other

Moreover, three service models were also reviewed, explaining their particular uses, advantages and disadvantages from an end-user and a security perspective. These models are (NIST, 2009):

- **Software as a Service** – A software application is deployed and managed by the Cloud provider, allowing the consumer to use this without needing to maintain or administer it
- **Platform as a Service** – Various services that make Cloud software development possible, without the consumer needing to setup or configure the underlying infrastructure
- **Infrastructure as a Service** – An infrastructure running on the Cloud, through which consumers can setup and run various applications or services, in a way that is comparable to a traditional computing environment

Finally, the Cloud was also defined in terms of four deployment models (NIST, 2009), highlighting the differences in the setup, level of control and level of security that these provide. These are:
**Private Cloud** – The Cloud infrastructure is used exclusively by a single consumer

**Community Cloud** – The Cloud is made available to a closed group of organisations

**Public Cloud** – The Cloud is made available to a much wider group of users

**Hybrid Cloud** – Two or more different Clouds linked together

Apart from defining the Cloud, its characteristics and various models, this study also proposed a taxonomy for Cloud use cases, which was inspired by the work of various other authors in this field. In doing so, the various types of Cloud implementations were organised in a way that was more manageable to understand, putting some structure around the vast uses of the Cloud. The taxonomy takes into consideration the various models and characteristics of the Cloud, subdividing these top level categories as necessary.

Based upon this material, the study progressed to propose five multi-dimensional conceptual models which are typical Cloud configurations that are often used in real-world implementations. For each model, the various actors, assets and data flows were defined, together with any assumptions done. These models were then used throughout the rest of the study to guide the discussion, applying the concepts being analysed to each of these typical configurations. The conceptual models considered are:

- **Model A** – A public PaaS model, which includes the use of third party actors
- **Model B** – A hybrid PaaS model, which assumes the existence of a specialised security team at the consumer organisation
- **Model C** – A public PaaS model
- **Model D** – A private IaaS model, hosted completely at the consumer organisation
- **Model E** – A public IaaS model

In order to meet the first objective set out in this study, the various benefits and concerns of the Cloud were also discussed. A recent study by KPMG found “the main drivers of Cloud Computing to be cost savings, improved flexibility and better scalability” (KPMG Netherlands, 2010), and these were reviewed in the light of the various Cloud models and characteristics. On the other hand, an analysis of the concerns involved in the Cloud found these to mainly revolve around security considerations. Three main risk classifications were defined in this study, namely data, management and provider related risks, and various examples were provided for each class of risk. Clearly, such risks need to be identified and analysed using an appropriate risk assessment methodology, and then addressed through relevant security controls in order to provide a level of security that is comparable to that of traditional computing.

Any risk assessment methodology requires the identification of the threats involved and, in this study, a number of threats were identified and discussed. Apart from the generic information security threats of confidentiality, integrity and availability, one also needs to consider Cloud-specific threats. These include:

- The availability of virtually unlimited Cloud compute power
- The vulnerability of Cloud management interfaces
- The possibility of Cloud provider insiders having malicious intents
- The fact that the Cloud infrastructure is shared by different tenants
- The lower levels of control that the user has over the data
- The fact that Cloud access control mechanisms can be attacked
- The threat of having an unknown risk profile
- The lack of sufficient forensic and auditing information in the Cloud
The applicability and degree of impact of these Cloud-specific threats depends on many factors, including the service and deployment models used. An analysis of these threats in the light of the conceptual models was also provided, in order to highlight these differences and to provide a more comprehensive threat analysis.

Apart from defining the threats involved, the study moved on to analyse the attitude that organisations have towards the Cloud, given the advantages, concerns and threats already discussed. This is essential since security is not only concerned with how effective the risk mitigation measures are, but it also takes into consideration how the users perceive security and the effectiveness of the security measures. In fact, “security is both a feeling and a reality and they are not the same” (Shneier, 2008). Various research was analysed in order to understand this user perception, with special emphasis done on scientific surveys carried out by (KPMG Netherlands, 2010) and (Ponemon Institute, 2012). It was determined that organisations are mainly concerned with security issues, legal considerations, compliance and privacy issues when using the Cloud. These perceptions, in fact, are sufficiently worrying to inhibit and limit the uptake of Cloud Computing. Such issues and the perceptions surrounding them need to, therefore, be addressed in order to encourage the use of Cloud Computing.

In order to address such security issues, a comprehensive Cloud risk assessment needs to be carried out, in order to methodically evaluate the risks involved. However, owing to the nature of the Cloud, such a risk assessment process is more complex than that of traditional IT systems. Various risk assessment frameworks and models have been put forward, which cover different parts of Cloud risk assessment. Based upon these, a Cloud Computing Risk Assessment Method (CC-RAM) is proposed in this study, which is a complete process for analysing and evaluating the Cloud’s risks. This cyclic model consists of eleven main phases, which are:

- **Asset Identification** – The data and process assets that fall within the scope of the risk assessment are identified
- **Asset Valuation** – The importance of these assets to the organisation is identified using any appropriate valuation methodology
- **Deployment Model Analysis** – The Cloud deployment model in question is considered, since this influences the risks involved
- **Service Model Analysis** – The Cloud service model used is also analysed, as this results in different information security threats
- **Data Flow Identification** – The data flows that are internal to the Cloud and those that flow in or out of the Cloud are identified
- **Threat and Vulnerability Analysis** – The particular threats and vulnerabilities that affect the assets are identified and analysed, using established and appropriate threat or vulnerability analysis models
- **Risk Score Calculation and Risk Treatment Planning** – A risk score is calculated based upon the previous findings, taking into consideration the impact, threat and probability, using risk matrices or other methodologies in the process
- **Countermeasure Identification, Gap Analysis and Control Implementation** – The countermeasures that are necessary to mitigate the risks are identified, and a gap analysis is then performed to compare these controls with what is already implemented. The missing controls are then implemented
- **Feedback Loops** – The risk assessment model should be regularly repeated, in order to ensure that any new or previously unhandled risks are identified and addressed as soon as possible
Cloud Encryption and Key Management Considerations

CC-RAM is based upon established standards and techniques, which are packaged together into a single model. This helps ensure that it can be easily adopted by organisations that already have other risk assessment procedures in place. Moreover, the model provides a certain degree of flexibility by allowing the user to incorporate various other, more specialised methodologies in each of its phases. This flexibility is essential since it should be possible for this model to be adopted by all Cloud players that need to perform a risk assessment.

9.2 Second Objective

To understand the role of encryption in the Cloud, and to outline challenges faced in its implementation.

Once a risk assessment is performed, a number of countermeasures need to be implemented in order to mitigate the risks. This study first proposes the use of cryptography in general as a suitable measure to implementing information security. Whilst cryptography on its own will not address all possible threats, it can be effectively used to address a number of security concerns, especially when used in combination with other countermeasures. Cryptography can be used to provide confidentiality and integrity, through various services, such as user authentication, data origin authentication, data integrity, non-repudiation and confidentiality. Mechanisms that implement these services include digital signatures, message access codes and encryption, to mention some common examples.

In particular, encryption can be used to implement confidentiality in a Cloud environment. Whilst encryption in itself does not prevent data from being lost or stolen, this makes it significantly more difficult for the data to be used by someone other than an authorised user. In fact, to a certain degree encryption can help address a number of threats that are faced in a Cloud environment. Referring to the main Cloud threats described earlier in this study, it was shown that encryption can help address various threats, including:

- **Data Security** – Encryption can be applied at different stages of the data’s lifecycle, to protect against confidentiality attacks such as malicious or accidental disclosure of confidential information
- **Cloud Management** – Encryption can protect not only the confidentiality of business data, but also that of Cloud-management messages and API calls
- **Cloud Provider Insiders** – Depending on the type of encryption and key management used, encryption can protect against unauthorised data access by Cloud provider insiders
- **Unknown Risk Profile** – Encryption can be used to mitigate the fact that the Cloud consumer might be unaware of various confidentiality threats that the data might be exposed to, once it is placed on the Cloud
- **Forensics** – Encryption can help ensure that the confidentiality of Cloud audit information is maintained, by ensuring that access to it is restricted on a need to know basis
- **Threats to Compliance** – Encryption can sometimes help achieve compliance with legislation, even in the case of unexpected events occurring

Moreover, this study also discussed how encryption can be applied to data that is at rest, in transit or in use. The choice of where encryption should be implemented depends upon the findings of a risk assessment, the level of risks involved, the organisation’s risk appetite and the particular Cloud model being used. For instance, the Cloud service and deployment model used influences where encryption should be best applied for this to be an effective security measure. In fact, an in-depth analysis of how encryption applies to each service model was also given in this study.
Encryption in a SaaS model can be implemented by various actors and at different points of the dataflow. This can be done by the consumer before the data is transmitted, by the provider once it is received, or by a third party while this is being transmitted. In PaaS, on the other hand, encryption can be done at two main levels, namely at the application and the Cloud infrastructure level. These types of encryption are managed by the consumer of the PaaS services and by the Cloud provider, respectively. Finally, encryption in IaaS can be volume-based, file-based or also at the application-level. Apart from describing each type of encryption and its suitability in achieving information security in the Cloud, these were also analysed with respect to the various types of Cloud infrastructure setups and use cases. The conceptual models proposed earlier in this study helped to achieve this, and to draw distinctions and similarities between the various types of encryption.

By analysing the work of other authors, it was noted how different organisations tend to prefer different types of encryption. Whilst this can be applied at various points of the data’s lifecycle, most organisations prefer to perform this before the data is placed on the Cloud. Moreover, organisations tend to be reluctant to entrust the Cloud provider storing the encrypted data, to also perform key management operations. Instead, these organisations prefer to be responsible for this task, or to entrust it to an autonomous third party.

However, irrespective of which actor performs encryption, and at which point in time this is done, the implementation and management of Cloud encryption presents various challenges and difficulties. These challenges are quite similar to those of encryption solutions in general, with the added complexity that the Cloud introduces. These can be classified into challenges which are inherently present in the encryption process, as opposed to those challenges which need to be addressed when implementing desirable (but non-essential) functionality in an encryption system.

Amongst the inherent challenges faced when implementing Cloud encryption, one needs to consider:

- **Encryption Policies** – The management and alignment of encryption policies becomes increasingly difficult as more entities are involved in the Cloud
- **Availability of Encrypted Data** – If Cloud data encryption is used, then ensuring the availability of this data and the related decryption services can be a challenge
- **Integrity of Encrypted Data** – Measures to ensure that the encrypted data is not tampered with need to be considered
- **Encryption Client Security** – Apart from considering the security aspects of the Cloud infrastructure, one needs to also evaluate those of the client machine
- **Compliance with Legislation and Standards** – Legal and compliance issues related to the storage of data and to its encryption also need to be addressed when dealing with encryption
- **Key Management** – One must also consider the related key management issues and how these can be addressed in order to ensure that the keys used do not get compromised

On the other hand, the challenges faced when implementing desirable functionality include:

- **Processing Encrypted Data** – It is desirable to be able to process data directly in its encrypted form, without compromising its confidentiality
- **Searching Encrypted Data** – Searching for data becomes significantly more complicated when this is encrypted, especially if the Cloud provider doesn’t know the decryption key
- **Sorting Encrypted Data** – Similar to searching, sorting is yet another challenge that is introduced when the data is encrypted
Cloud Encryption and Key Management Considerations

- **Encryption Interoperability** – Interoperability is desirable since it enables the encryption and decryption processes to be performed by various entities and on different devices.

With any encryption system, one also needs to consider the associated key management services. Key management is often the Achilles’ heel of an encryption system and it should, therefore, be carefully considered. When implementing Cloud key management, one also encounters a number of challenges, including:

- **Key Generation and Storage** – Generation of keys used during encryption needs to be done in a secure manner, ensuring that these keys are strong and stored securely.
- **Key Availability** – Once keys are generated and used to encrypt data, these need to be available whenever needed, to ensure that the data they protect can be accessed.
- **Key Disposal and Expiration** – The disposal and revocation of keys when these are no longer required is a complex task that needs to be addressed.
- **Key Management Policy** – Key management processes should be part of a wider security policy, and it must be consistent or complementary across the different players involved in Cloud encryption.
- **Separation of Duties** – Key management processes should not be provided by an entity other than that storing the encrypted data, to protect against threats to confidentiality, integrity and compliance.
- **Key Management Interoperability** – Key management interoperability is a challenge since different providers and encryption algorithms typically have different key management requirements.

It is, therefore, challenging to securely implement encryption and key management in a Cloud environment. Moreover, as discussed in this study through the use of the conceptual models, the difficulty in addressing these challenges depends upon the particular Cloud model being used. Similarly, the degree of impact that these have on the level of information security in the Cloud is also affected by the type of Cloud setup that is being adopted.

In order to further understand what role encryption plays in the Cloud, this study proceeded to also consider this from a legal perspective. In the Cloud, different laws and directives come into play, covering sectors such as financial, health care, government and privacy. In this study, the EU Data Protection Act and the USA Health Insurance Portability and Accountability Act were discussed as two such examples. Complying with these laws is often already difficult in a traditional business environment, and this is made substantially more complicated in the Cloud. This is because the data does not remain under the sole responsibility of the organisation, but it is entrusted to third parties on the Cloud, who will often process the data and also transmit or store this in data centres found in different legal jurisdictions. Whilst the use of encryption is sometimes mandated by law, in other cases this can help achieve compliance by ensuring that the confidentiality of the data is maintained even when this is placed on the Cloud.

Despite this, the use of encryption in itself should not provide a false sense of complete security, since issues and confidentiality breaches might still occur. The use of information security standards should, therefore, be considered when implementing Cloud encryption, in order to ensure compliance with best practices and to avoid common pitfalls that might unnecessarily reduce the degree of security provided. This study considered the use of generic standards such as ISO27001, ISO27002, ISO27005, BS25999 and COBIT, and how Cloud encryption and key management relate to such standards. Moreover, Cloud-specific security standards also exist, such as the OSA Cloud Computing Pattern. Such a pattern can be used to guide the secure implementation of encryption and key management in the Cloud and, as described in this study,
the Cloud Computing Risk Assessment Method proposed earlier can also be effectively incorporated into this pattern. Such Cloud-specific standards are, however, still in their infancy and plenty of work is still being done in this area.

9.3 Third Objective

To review encryption and key management approaches that aim to address the Cloud security issues

Implementing Cloud encryption and key management is, therefore, not without its challenges. However, standards and patterns can help guide their implementation to ensure that these are done in a recommended way. Based upon these standards, as well as the analysis and discussions done earlier in this study, a number of best practices and proposals were derived for both key management and encryption.

Key management is an essential component of Cloud encryption and, therefore, all necessary measures should be taken to ensure that this is implemented in a secure and robust manner. These include:

- **Use of Key Management Standards** – Cloud key management should be based upon established standards and protocols, which cover specific aspects of key management such as storage, availability, retrieval, sharing and revocation of keys. Standards that were reviewed in this study include NIST SP800-57, ISO 11770-1, IEEE 1619.3 and OASIS KMIP.

- **Key Generation and Storage** – Keys should be generated using adequate techniques which include good random number generators. Moreover, keys should be generated for a single-purpose, to avoid unnecessarily compromising them by using the same key for different purposes. Once generated, the keys should then be stored in a secure manner, possibly protected using cryptographic techniques. An important decision that needs to be done is which Cloud actor will be responsible for storing and maintaining these keys, and the use of Split Key Management was discussed as a possible solution to this problem.

- **Key Availability** – Mechanisms should be in place to ensure that the keys can still be retrieved in the event of loss or damage to the primary key storage service, or to the mechanisms that protect the keys. The use of key escrow was discussed as a suitable way of meeting this recommendation.

- **Key Distribution** – Measures should be taken to ensure the keys’ confidentiality, integrity and authenticity during distribution. Moreover, techniques should be adopted that facilitate the provision of the necessary keys to those that are authorised to obtain a copy. The use of Federated Key Management was discussed as a possible way of simplifying these problems, by allowing multiple actors to directly communicate with the same key management provider.

- **Key Termination** – Keys should have a predefined lifetime, after which they should be terminated. Any data that is protected by these keys should be either disposed of, or translated to use a new key.

- **Key Management Policy** – A key management policy should be established as part of a complete policy framework adopted by the organisation. Through an effective policy, the organisation can ensure that key management processes are better understood and easier to follow by all entities involved in the process. Standards such as SP800-57 and SP800-131A should be used to guide the establishment of such policies.
• **Separation of Duties** – Key management services should not be provided by the same entity that is providing the storage services. Typically, storage is provided by the Cloud provider, in which case the key management services should be managed by the Cloud consumer or by a third party.

• **Key Management Interoperability** – Standard key management techniques should be used, since these provide interoperability. Through interoperability, a single key management system can be used by multiple entities and applications. The use of the Key Management Interoperability Protocol was discussed as a possible way of achieving this interoperability.

• **Human Aspect of Key Management** – Access to and use of the keys should be controlled, restricted and audited. This makes it more complicated for users to maliciously gain access to and use the keys, and it also serves as a deterrent. Moreover, auditing will provide the necessary forensic information in the case of a malicious use.

The study reviewed the applicability of these best practices and proposals with respect the conceptual models defined earlier, in order to evaluate their relevance in the different Cloud setups that might be encountered. Similarly, the study also proposed a set of best practices for Cloud encryption, which are:

• **Encryption Policy** – An encryption policy should be in place, which covers the entire Cloud encryption process. This policy should be supported using all appropriate technological and procedural controls.

• **Encryption Algorithms** – The encryption algorithm to be used should be carefully selected, since different encryption techniques have different advantages and disadvantages. Symmetric and asymmetric key encryption were discussed as two such examples.

• **Integrity of Encrypted Data** – Mechanisms that enable the Cloud consumer to verify the integrity of the ciphertext, should be adopted. These help ensure that the data has not been tampered with. The use of the Proof of Storage protocol was discussed as a possible way of achieving this integrity.

• **Encryption Client Security** – The client machine running the encryption process and consuming the keys should be secured and protected, to ensure that this is not vulnerable and compromised.

• **Legislation and Compliance** – Encryption algorithms and keys used should be selected such that these ensure compliance with legislation across the different jurisdictions where the Cloud data might be stored or processed.

• **Encryption that Enables Easier Management** – Cloud encryption should leverage upon encryption techniques that simplify the management processes. Identity Based Encryption and Attribute Based Encryption were discussed as two such examples. The former simplifies key distribution, while the latter allows for access policies to be inbuilt directly into the encryption process.

• **Encryption Interoperability** – Algorithms used should be standard and interoperable, allowing the encryption or decryption process to be performed by any entity who has access to the necessary keys.

• **Implementation of Desirable Features** – Encryption should be considered to be an enabler rather than a limiting factor. For this reason, techniques that allow encrypted data to be processed in a similar manner to plaintext data, should be investigated and adopted where possible. Two such examples are the ability to process encrypted data directly, as well the ability to search and sort this data. This study discussed the use of Homomorphic Encryption and Searchable Encryption as possible ways of achieving this
aim, highlighting the fact that these are still theoretical approaches that are not yet ready for real world implementations.

9.4 Future Work and Concluding Remarks

Apart from providing a comprehensive review and discussion on Cloud encryption and key management, this study also proposed taxonomies, models and best practices. However, there is still room for a significant amount of research to be done in this field. Most of this future work relates to areas which are currently unexplored, or which stand to benefit from further analysis. Whilst this study evaluated some Cloud-specific standards, a more in-depth and comprehensive study of such standards would be beneficial. Unfortunately, such standards are currently still in their infancy and not sufficiently refined, so an analysis is needed to identify how such standards relate to one another, and what the similarities or discrepancies between these models are. Moreover, techniques such as Fully Homomorphic Encryption and scalable Searchable Encryption are still not ready for actual use, and further research should be done in these areas to improve upon these techniques. Further work can also be done on the Cloud Computing Risk Assessment Method proposed in this study, which can be refined further by developing a sub-model for each of its eleven stages and formalising the risk assessment model even further.

Whilst there is still room for more research to be done in the field of Cloud encryption, the objectives that were set out in the beginning of this study were achieved. Although it is very unlikely that all threats, risks, challenges and best practices proposed in this study will apply for any single Cloud scenario, at least some of these will apply to any Cloud use case. As an example, a case study on Dropbox was given, which related the applicable material of this research to a Cloud file storage application. In doing so, ways of securely implementing encryption in this application were discussed, which aim to improve the application’s information security levels. It is essential that measures such as encryption are properly implemented whenever necessary, since these can help improve security. Trends show that Cloud security awareness and measures are constantly improving and, as discussed in this study, security will soon change from being a Cloud inhibitor to being one of the key enablers of Cloud adoption (Penn, 2010).
Bibliography


Cloud Encryption and Key Management Considerations


Cloud Encryption and Key Management Considerations


Cloud Encryption and Key Management Considerations


Monaco, A. (2012). *A View Inside the Cloud*. IEEE.


OASIS. (2012). *KIMP Cloud Use Cases*. OASIS.


Cloud Encryption and Key Management Considerations


Cloud Encryption and Key Management Considerations


Appendix A – Security Effectiveness Score

This list is reproduced from (Ponemon Institute, 2007)

The following 24 attributes are used to describe an effective IT security based on responses from survey participants. These attributes comprise an organization’s Security Effectiveness Score (SES).

1. Identify major data breaches involving sensitive or confidential information
2. Determine the root causes of major data breaches involving sensitive or confidential information
3. Know where sensitive or confidential information is physically located
4. Secure sensitive or confidential data at rest
5. Secure sensitive or confidential data in motion
6. Secure endpoints to the network
7. Identify system end users before granting access rights to sensitive or confidential information
8. Protect sensitive or confidential information used by outsourcers (including third parties, affiliates, and business partners)
9. Prevent or curtail major data breaches involving sensitive or confidential information
10. Prevent or curtail hacking attempts to acquire sensitive or confidential information
11. Prevent or curtail denial-of-service attacks
12. Limit physical access to data storage devices containing sensitive or confidential information
13. Demonstrate the economic value or other tangible benefits of the company’s IT security program
14. Ensure minimal downtime or disruptions to systems resulting from security problems
15. Comply with legal requirements and policies (including privacy laws and statutes)
16. Conform with leading self-regulatory requirements such as ISO 17799, PCI, and others
17. Prevent or curtail viruses, worms, Trojans, and spyware infections
18. Perform timely updates for all major security patches
19. Control all live data used in systems development activities
20. Enforce corporate policies, including the termination of employees or contractors who pose a serious insider threat
21. Attract and retain high-quality IT security personnel
22. Training and awareness program for all system users
23. Conduct independent audits of the system
24. Consistently manage security program administration