Department of Technologies and Information Systems

University of Castilla-La Mancha

PhD Thesis

SMiLe2Cloud - Security Migration of Legacy systems to Cloud computing

PhD Candidate: Luis Márquez Alcañiz
Supervisors: Dr. David García Rosado
           Dr. Haralambos Mouratidis
           Dr. Eduardo Fernández-Medina Patón
Department of Technologies and Information Systems

University of Castilla-La Mancha

PhD Thesis

SMiLe2Cloud - Security Migration of Legacy systems to Cloud computing

Ciudad Real (Spain), November 2017

PhD Candidate: Luis Márquez Alcañiz
Supervisors: Dr. David García Rosado
Dr. Haralambos Mouratidis
Dr. Eduardo Fernández-Medina Patón
Dedication:

To my family, and especially to my lovely wife, my main motivation to achieve my goals.
Acknowledgments

First and foremost, I would like to express my sincere gratitude to my advisors, Dr. David García Rosado, Dr. Eduardo Fernández-Medina and Dr. Haralambous Mouratidis for their confidence, support, demands and expert guidance. They have made this work possible. The assistance they provided in the preparation of this manuscript has been incalculable.

I would like to thank the people who work or have worked with me during these years at the GSyA Research Group for their invaluable help and advice received during the preparation of this thesis: thanks to Goyi and Marisa because they make things easier and especially thanks to Dr. Ignacio García-Rodríguez de Guzmán for his help in the initial stages of this thesis. Besides, I would like to thank the members of the SenSe group of the University of Brighton who have helped and enriched me with their comments and advice and especially Nikos and Shaun who has collaborated on this thesis.

I would like to thank my colleagues from the CNMC, for his support and dedication to me and my work.

At last but not least, I would like to thank my family for their love and support.
If you think technology can solve your security problems, then you don't understand the problems and you don't understand the technology.

Bruce Schneier
Contents
# Table of Contents

Contents.............................................................................................................................................. I

Table of Contents ................................................................................................................................ III

List of Figures ........................................................................................................................................ VII

List of Tables ....................................................................................................................................... XI

Abstract............................................................................................................................................... 1

1. INTRODUCTION ............................................................................................................................... 3

1.1. MOTIVATION .................................................................................................................................. 5

1.2. CLOUD COMPUTING .................................................................................................................... 6

1.2.1. Cloud Computing Features ...................................................................................................... 7

1.2.2. Classification of Cloud Computing Models .............................................................................. 8

1.2.3. Advantages of Cloud Computing ............................................................................................. 9

1.2.4. Cloud Computing Forecasts and Market Estimates ................................................................. 10

1.3. SECURITY IN CLOUD COMPUTING .......................................................................................... 13

1.3.1. Introduction to Information Security ......................................................................................... 13

1.3.2. Risks Associated with Cloud Computing .................................................................................. 14

1.4. LEGACY SYSTEMS MIGRATION TO CLOUD ........................................................................... 15

1.4.1. Legacy Systems .......................................................................................................................... 15

1.4.2. Legacy Systems Migration ......................................................................................................... 16

1.5. HYPOTHESIS AND GOALS .......................................................................................................... 19

1.6. PHD THESIS FRAMEWORK ....................................................................................................... 19

1.6.1. Research Group ......................................................................................................................... 20

1.6.2. R&D Projects ............................................................................................................................. 20

1.7. DOCUMENT OUTLINE .................................................................................................................. 21

2. RESEARCH METHOD ....................................................................................................................... 23

2.1. ACTION-RESEARCH METHOD .................................................................................................... 25

2.1.1. Definitions .................................................................................................................................. 25

2.1.2. Roles and Variations in the Action-Research ............................................................................. 26

2.1.3. Stages of Action-Research ......................................................................................................... 27

2.1.4. Problems of Action-Research and Alternatives ......................................................................... 28

2.1.5. Action-Research Cycles .............................................................................................................. 29

2.1.6. Application of Action-Research to this PhD Thesis .................................................................. 30

2.2. SYSTEMATIC LITERATURE REVIEWS ..................................................................................... 33

2.2.1. Method Used to Conduct a Systematic Literature Review ....................................................... 33

2.2.2. Systematic Mapping Studies ..................................................................................................... 37

2.2.3. Application of Systematic Literature Reviews to this PhD Thesis ........................................... 38

3. STATE OF THE ART .......................................................................................................................... 41

3.1. CLOUD INFORMATION SECURITY SYSTEM PROPOSALS ....................................................... 43

3.1.1. Standards and Reference Documents ......................................................................................... 43

3.1.2. Security Methodologies for the Development of Information Systems .................................... 48

3.1.3. Proposals for Cloud Computing Security .................................................................................. 55

3.2. PROPOSALS REGARDING THE MIGRATION OF LEGACY SYSTEMS TO THE CLOUD ............. 62

3.2.1. Methodologies for the Migration of Legacy Systems ................................................................. 62

3.2.2. Methodologies for Migration to the Cloud ................................................................................. 67

3.2.3. Methodologies for Secure Migration to the Cloud ..................................................................... 72

3.3. COMPARISON OF PROPOSALS ............................................................................................... 77
List of Figures

Figure 1-1. Size of the hosting and cloud computing market from 2011 to 2019 .......... 11
Figure 1-2. Estimated Public Cloud Revenue ............................................................... 11
Figure 1-3. Worldwide Cloud IT Infrastructure Market Forecast ............................... 12
Figure 1-4. Microsoft Commercial Revenue Growth ............................................... 12
Figure 1-5. Horseshoe model for legacy software migration ................................... 17
Figure 2-1. Action-Research Process ................................................................. 28
Figure 2-2. Two Dimensions for Information Systems Action-Research ......................... 30
Figure 2-3. Application of Action-Research: Participants ........................................... 32
Figure 2-4. Publications in Google Scholar ............................................................... 39
Figure 2-5. Evolution of publication in Google Scholar ............................................. 40
Figure 3-1. CSA Domains ......................................................................................... 44
Figure 3-2. Method used to build a SRA for clouds .................................................. 58
Figure 3-3. SOAMIG Core disciplines ......................................................................... 64
Figure 3-4. MARBLE, a framework to support business process archaeology .............. 66
Figure 3-5. Process model refactoring algorithms introduced by IBUPROFEN ............... 67
Figure 3-6. CloudSDLv1 phases ................................................................................ 74
Figure 3-7. Five-Phase-Model of secure migration to a Cloud Service Provider ............ 75
Figure 4-1. SMiLe2Cloud (smiletocloud.com) .......................................................... 83
Figure 4-2. The SMiLe2Cloud activities ...................................................................... 84
Figure 4-3. SMiLe2Cloud roles represented according to SPEM 2.0 ......................... 86
Figure 4-4. Notation used to model the activities and tasks according to SPEM ............. 88
Figure 4-5. Secure Tropos: Main Concepts ............................................................... 92
Figure 4-6. SMiLe2Cloud model concepts .............................................................. 93
Figure 4-7. SMiLe2Cloud sub artefacts ....................................................................... 94
Figure 4-8. The SMiLe2Cloud model ......................................................................... 96
Figure 4-9. SPEM 2.0 view of Inputs, Outputs and Stakeholders of the extraction activity 97
Figure 4-10. SPEM 2.0 view of tasks and artefacts of the extraction activity .................. 98
Figure 4-11. SPEM 2.0 view of Inputs, Outputs and steps of Task 1.1 ......................... 99
Figure 4-12. Detailed description of the task 1.1 using SPEM 2.0 ................................. 100
Figure 4-13. SPEM 2.0 view of Inputs, Outputs and steps Task 1.2 ......................... 101
Figure 4-14. Detailed description of Task 1.2 using SPEM 2.0 ................................. 102
Figure 4-15. SPEM 2.0 view of Inputs, Outputs and steps of the task 1.3 ................. 103
Figure 4-16. Detailed description of the task 1.3 using SPEM 2.0 ............................... 104
Figure 4-17. SPEM 2.0 view of Inputs, Outputs and Stakeholders of the analysis activity 105
Figure 4-18. SPEM 2.0 view of tasks and artefacts of the analysis activity ................... 106
Figure 4-19. SPEM 2.0 view of Inputs, Outputs and steps of Task 2.1 ....................... 107
Figure 6-8. REM SMiLe2Cloud goal model ................................................................. 151
Figure 6-9. REM SMiLe2Cloud goal model in XML .................................................... 152
Figure 6-10. SecTro tool ............................................................................................ 153
Figure 6-11. REM SMiLe2Cloud security requirements ............................................... 154
Figure 6-12. REM SMiLe2Cloud cloud controls ......................................................... 155
Figure 6-13. Degree of compliance of Amazon and Azure ........................................... 158
Figure 6-14. Technological SWOT analysis of Microsoft Azure and Amazon AWS ....... 160
Figure 6-15. Interface deployed in Amazon ................................................................. 163
Figure 6-16. Infrastructure deployed in Amazon .......................................................... 163
Figure 6-17. CNMC applications employed in order to comply with Law 39/2015 .......... 170
Figure 6-18. Current WECO architecture ................................................................. 172
Figure 6-19. WECO Interface ................................................................................... 173
Figure 6-20. WECO SMiLe2Cloud goal model ........................................................... 174
Figure 6-21. REM SMiLe2Cloud security requirements .............................................. 175
Figure 6-22. Minimum WECO architecture ............................................................... 180
Figure 6-23. Deployment infrastructure in Azure ....................................................... 181
Figure 6-24. JSON deploy server to Azure ................................................................. 182
Figure 6-25. WECO in Azure ..................................................................................... 182
List of Tables

Table 1-1. Projects supported by this PhD Thesis ....................................................................... 20
Table 3-1. Summary of proposals analysed ................................................................................ 77
Table 4-1. Global schema of the SMiLe2Cloud activities and tasks ............................................ 87
Table 4-2. Security goals ............................................................................................................. 90
Table 4-3. Algorithm for the transformation process ............................................................... 103
Table 4-4. Security controls aligned with RMIAS ..................................................................... 108
Table 4-5. Capability levels combined for security verification ................................................ 124
Table 6-1. Set of controls applicable to the case study ............................................................ 156
Table 6-2. Physical infrastructure of REM ................................................................................. 159
Table 6-3. Comparison of costs.................................................................................................. 159
Table 6-4. Security mechanism categories ............................................................................... 164
Table 6-5. Results for secure login category ............................................................................. 165
Table 6-6. Results for certificate management category .......................................................... 166
Table 6-7. Results for secure channel category ........................................................................ 166
Table 6-8. Results for secure channel category ........................................................................ 167
Table 6-9. Set of controls applicable to this case study ............................................................ 176
Table 6-10. Physical infrastructure of WECO ............................................................................ 178
Table 6-11. Comparison of costs............................................................................................... 178
Table 6-12. Security mechanism categories ............................................................................. 183
Table 6-13. Security mechanism categories ............................................................................. 183
Table 7-1. Summary of publications achieved .......................................................................... 193
Table 7-2. Publications classified by contributions of thesis ....................................................193
Table 7-3. Publications in international conferences ................................................................. 194
Table 7-4. Publications in international conferences ................................................................. 195
Table 7-5. Publications in international journals ......................................................................... 195
Table 7-6. Publications in national journals ............................................................................... 195
Table 7-7. List of cites in publications ....................................................................................... 196
Cloud computing has become a fundamental part of the technological strategy of companies and is one of the hottest core technical topics in the modern era. Cloud computing offers organisations many potential benefits by making information technology services available as a commodity. Cloud Computing allows companies to offer better services by reducing costs through improved utilisation, reduced administration and infrastructure costs, faster deployment cycles and the ability to respond more quickly and reliably to their customers’ needs. The cloud computing market is expected to reach 270 billion dollars by 2020.

The adoption of cloud computing in order to provide the platform, infrastructure and services for the multitenant has grown exponentially during recent years. However, despite its various advantages, cloud computing also fosters security concerns that hamper its rapid adoption. Security is one of the major issues that are reducing the growth of cloud computing, and complications with data privacy and data protection continue to plague the market. The transformation of local computing into remote computing has brought with it many security issues and challenges for both the consumer and the provider, because many cloud services are provided by the trusted third party which provides its services through the Internet and uses many web technologies that lead to new security issues.

The move to cloud computing is a business decision that takes into account the readiness of existing applications for cloud deployment, the transition and life-cycle costs, the maturity of service orientation in the existing infrastructure, and the organisation’s security and privacy requirements. Cloud computing systems are complex networked systems that are affected by traditional computer and network security issues, such as the need to provide data confidentiality, data integrity, and system availability.

The objective of legacy system migration is to move operational systems towards new platforms and retain their original functionality whilst simultaneously minimising disruption to the operational and business aspects. A considerable number of companies are interested in migrating their legacy systems to the cloud, but there is a need to do so in an appropriate manner, i.e. ensuring that the security of the systems to be migrated is maintained and improved. Security, or rather the lack thereof, is the main concern for 69% of companies when migrating.

There are various studies, existing international standards and “best practice” documents covering the area of security. However, much still remains to be done before these studies and standards cover the full complexity of the cloud computing model. Various studies propose different methodologies for the migration of legacy systems and there are even several methodologies with which to migrate legacy systems to cloud computing. There are also several proposals regarding security in cloud computing, however, an initiative for the migration process of the security aspects from legacy systems to cloud computing is not currently available.

In this PhD Thesis, we propose a framework, the SMiLe2Cloud framework, with which to support the secure migration of legacy systems to cloud computing. That is, the SMiLe2Cloud framework will support the migration process of legacy systems in order to verify that the resulting system in cloud computing is secure.
The objective of the SMiLe2Cloud framework is not only to fulfil the security requirements of the legacy system in the cloud but also, and this is its real objective, to improve the security of legacy software in this new environment.

The SMiLe2Cloud framework defines the activities, tasks and steps to be performed, the elements used in each task, the artefacts used in each activity and the people responsible involved.

The SMiLe2Cloud framework covers the entire life cycle of the migration process. It begins with the extraction of legacy system requirements, continues with the analysis, design and implementation of the security requirements, and finally evaluates the entire process to ensure that all the objectives have been achieved. The framework is based on a Deming cycle of continuous improvement, and a continuous re-evaluation of possible improvements to the system will, therefore, be carried out.

SMiLe2Cloud is led by standards, signifying that its proposed activities are aligned with the current best practices. The perspective followed in our approach has been modelled following the SPEM 2.0 (Software & Systems Process Engineering Meta-Model) specification, thus facilitating its inclusion in any organisation.

Besides the definition of SMiLe2Cloud framework, an ecosystem of tools is suggested, in order to facilitate the migration process. The ecosystem provides a set of tools in an integrated and coordinated manner so as to provide support for the proposed methodology, from the extraction activity to the evaluation activity.

Furthermore, and with the purpose of validating the practical applicability of this framework and improve its development, the results of two full practical empirical evaluation in a real organisation are provided.
1. Introduction
1.1 Motivation

This chapter provides a description of the general aspects related to this PhD thesis. First, in Section 1.1, the motivation and importance of the work is presented. In Section 1.2, an introduction to cloud computing is put forward, while the main aspects of security in cloud computing are explained in Section 1.3. In Section 1.4 the principal aspects of legacy systems migration are described. The hypothesis and pursued goals are presented in Section 1.5 with the aim of facilitating the reading and comprehension of this thesis. Section 1.6 shows an explanation of the framework of the thesis, with reference to R&D projects. Finally, in Section 1.7, the general organisation of this PhD thesis is described.

1.1. Motivation

Cloud computing is currently a major trend in the IT industry, and is one of the main options to consider when outsourcing services. The many advantages offered by this computer service model have attracted the interest of a considerable number of companies that wish to adopt them.

In order to really understand the level of expectation placed upon cloud computing, one only has to read the report published by the European Commission entitled "Unleashing the Potential of Cloud Computing in Europe" (European Comission, 2012). This report anticipates that the potential impact of cloud computing could result in "a net gain of 2.5 million new European jobs, and an annual boost of €160 billion to the European Union GDP (Gross Domestic Product), by 2020". Another more recent study from the European Commission entitled “Uptake of Cloud in Europe” (European Commission, 2015) has found that, from 2015 to 2020, cloud computing could add a cumulative total of €449 billion to the European Union’s GDP.

It has been claimed that cloud computing takes enterprises’ search to a new level and allows them to further reduce costs through improved utilisation, reduced administration and infrastructure cost and faster deployment cycles (Kushwah & Saxena, 2013).

In (Meritalk & Dynamics, 2016), it is stated that 85% of Federal IT managers are currently more optimistic about what the cloud can do for their agencies than they were five years ago. Federal IT managers are also becoming more comfortable with Cloud Service Providers (CSPs). In 2014, 44% of Federal IT managers said they were somewhat or very comfortable about turning their data over to a cloud provider, and that number has now increased to 79%.

In the “Future Ready Applications” report (Meritalk & Accenture, 2015), MeriTalk surveyed 150 Federal IT managers to discover how application modernisation can breathe new life into legacy apps and deliver much-needed efficiency and security. An overwhelming majority of Feds agreed that it is time to modernise, while 92% of Federal IT managers stated that it is urgent for their agency to modernise legacy applications. The driving factors include security issues (42%), the time required to manage/maintain systems (36%), inflexibility (31%), and integration issues (31%). However, and despite the urgent need for modernisation, only half of the agencies in question (53%) have a formal application modernisation strategy.

The essence of legacy system migration is to move an existing, operational system to a new platform, retaining the functionality of the legacy system while causing as little disruption to the existing operational and business environment as possible (Wu et al., 1997). Legacy system migration is a very expensive procedure which runs a definite risk of failure. Consequently, before any decision to migrate is taken, an intensive study should be undertaken to quantify the risks and benefits and fully justify the redevelopment of the legacy system involved (Bisbal, Lawless, Wu, & Grimson, 1999; Bisbal et al., 1997).
IT executives have, from the very beginning, expressed their principal concern: security. This was so in the first years after the concept emerged (Christiansen, Kolodgy, Hudson, & Pintal, 2010; Gens, 2008; The Open Group, 2011), and it continues to be so today (Netwrix Corporation, 2015). The Netwrix 2015 Cloud Security Survey found that 65 percent of companies still have security concerns when migrating to the cloud, while 40 percent are still worried about their loss of physical control over data in the cloud. Security plays a recurring role in the concern about cloud adoption, as do concerns about data privacy.

Security consistently raises most questions as consumers look to move their data and applications to the cloud, although cloud computing does not imply any security issues that have not already been raised for general IT security. The concern when moving to the cloud is that implementing and enforcing security policies now involves a third party. This loss of control emphasizes the need for transparency from cloud providers (Ahronovitz et al., 2010).

Other people, however, see in the cloud computing paradigm a good chance to improve the security of legacy software (Aikat et al., 2017; Chang & Ramachandran, 2016; Rittinghouse & Ransome, 2016). For instance, Winkler states in (Winkler, 2011) that the migration from legacy systems to the cloud “gives us hope that we can regain control [...] from poorly integrated or after-thought security.” In some cases, the cloud will offer a better position as regards security than an organisation could otherwise provide.

Given that security is mostly questioned by very same practitioners that will have to decide, the apparent absence of an acknowledged migration model for legacy systems with specific security concerns could deter the widespread adoption of the model. This would thus appear to be a valuable opportunity for research.

The goal of the thesis is, therefore, to propose a framework with which to support the security migration of legacy systems to cloud computing in the form of a set of methods that address the issue of security and how security should be integrated with different kinds of processes in order to migrate legacy information systems to the cloud.

1.2. Cloud Computing

The objective of this section is to highlight the principal characteristics of cloud computing, along with identifying how this type of services can be classified, in order to better understand its functioning.

The definition of cloud computing that is most widely accepted by the scientific and technical community is that provided by the NIST (National Institute of Standards and Technology), which states that cloud computing is a computation model that permits access on demand via the Web, from anywhere and in a practical manner, to a shared set of configurable computing resources (e.g. networks, servers, storage, applications and services), which can be rapidly provisioned and freed with the minimum management effort or interaction on the part of the service provider (Mell & Grance, 2011).

This perspective is complemented by that of the University of Berkley, which states that cloud computing refers both to the applications and services offered via the Internet, such as the hardware and software systems that data centres make available in order to provide those services (Armburst et al., 2009). The hardware and software at these data centres constitute what has been denominated as the cloud. The aforementioned authors state that the principal innovations provided by this paradigm are the following:
1.2 Cloud Computing

- The illusion of infinite computing resources available on demand.
- The absence of a closed-capacity commitment, thus making it possible to begin with few resources and increase if and when necessary.
- The ability to pay exclusively for the use of the computing resources used.

All of the existing definitions of cloud computing share a set of common concepts, according to which it is possible to define this type of services as a large set of easily accessible and usable virtualised resources, which can be dynamically configured in order to adjust to a variable load, thus permitting an optimum use of those resources. This set of resources is typically exploited in a pay for use model in which the provider provides guarantees by means of personalised service level agreements (SLAs) (Vaquero, Rodero-Merino, Caceres, & Lindner, 2009).

With these definitions it will be appreciated that the cloud computing paradigm supposes an important change for organisations as regards managing Information Technologies (IT). This change makes it possible for companies to move from the use of a traditional management model in which infrastructures are directly acquired and managed, to the new model in the cloud in which services are contracted from the provider.

1.2.1. Cloud Computing Features

The cloud computing model has its own set of characteristics that make it different from traditional IT management models:

- Self-service on demand. Any customer can automatically be supplied with computing capacity, such as server time or storage in the Web, according to his or her requirements and without the need for human intervention with each service provider.
- Unlimited multi-platform access. These capabilities can be accessed via the Web by means of standard mechanisms that promote their use from any type of platform.
- Shared resources. The provider’s computing resources are shared by multiple customers in a rent-by-use model, and different physical and virtual resources are dynamically assigned and reassigned according to those customers’ demands. There is a sense of independence from location in that the customer does not normally have control over or knowledge of the exact location of the resources provided, but may require the ability to specify a delimited location, such as a country or state.
- Scalability. This capacity can be supplied in an elastic manner in order to rapidly scale up or decrease according to the demand. From the customer’s point of view, the capacity available for supply appears to be unlimited and can be requested in any quantity and at any time.
- Measurement of service. Cloud computing systems automatically control and optimise the use of resources by measuring their use in accordance with the type of service provided. The use of resources can be monitored and controlled, thus providing both the provider and the customer of the cloud service with transparency.
1. Introduction

- Abstraction. The service provider’s capacity to isolate the computing services contracted, resulting in transparent management for the customer. This ensures that the customer does not require personnel to maintain the infrastructure.

- Pay for use. This is derived from scalability, and the cost of the service is modified according to the use that the customer makes of the resources. The cost may, therefore, vary according to the customer’s specific momentary needs.

1.2.2. Classification of Cloud Computing Models

The various alternatives offered by the cloud computing service market can be classified on the basis of two criteria: the service model and the deployment model. The various classifications provided by (Mell & Grance, 2011) are explained as follows.

1.2.2.1. Classification of Service Models

This criterion refers to the level of abstraction of the service offered by the provider. The principal typologies are:

- Software as a Service (SaaS). The provider offers the customer the capacity to use applications that are executed in an infrastructure in the cloud. These applications can be accessed from multiple platforms by customers who are both heavyweight owners and lightweight Net surfers. The customer neither manages nor controls the underlying infrastructure (networks, servers, operative systems, storage or applications), with the exception of possible adjustments to the applications they handle.

- Platform as a Service (PaaS). The provider offers the customer the capacity to deploy applications that s/he has created or acquired in the cloud infrastructure using programming languages, libraries, services and tools supported by the provider. The customer does not manage the underlying infrastructure, but does control the applications deployed and the possible adjustments to the configuration of the environment.

- Infrastructure as a Service (IaaS). The provider offers the capacity to provide computation resources (processors, storage, network, etc.) in which the customer can deploy the software of his/her choice, including operative systems and applications. The customer controls the operative systems, the storage and applications installed. S/he may possibly have limited access to certain elements of the Net, such as some firewalls between servers.

1.2.2.2. Classification of Deployment Models

This criterion refers to the sharing of resources contracted with the provider with other entities and customers of a similar or different nature. The principal typologies are:
1.2 Cloud Computing

- **Private cloud.** The cloud computing infrastructure is provided for the exclusive use of one single organisation, although this organisation may have multiple business units that act as customers. It can be managed and operated by the organisation itself or by a third provider party, and can be located in the customer organisation itself or in the exterior.

- **Community cloud.** The cloud computing infrastructure is provided for the specific use of a community of customers with some sort of common interest, such as a mission, politics or common considerations. It can be managed and operated by one of the organisations from the community or by a third provider party, and can be located at the community’s headquarters or in the exterior.

- **Public cloud.** The cloud computing infrastructure is provided for the use of the general public. It can be managed and operated by a company or by academic or governmental organisations, and is located at the provider’s headquarters.

- **Hybrid cloud.** The cloud computing infrastructure is composed of the aforementioned models (private, community and public), such that they are maintained as separate entities but interact by means of a predefined technology that allows the portability of data and applications.

1.2.3. **Advantages of Cloud Computing**

The interest that cloud computing is provoking as regards being an IT service provision model is owing to its numerous advantages. The various benefits of this new paradigm are summarised as follows.

- **Benefits related to setting the service in motion.** IT projects are normally based on an infrastructure that must be deployed, thus guaranteeing interoperability with other technological components. Cloud computing facilitates this deployment and reduces the time needed to get the service going (Deloitte, 2009).
  
  - **Immediateness:** a productive service can be made available in a very short amount of time, thus shortening the initial stages of the IT project.
  - **Initial investment:** it is not necessary to acquire fixed assets or to equip data centres.
  - **Comfort:** users can use the new service without having to update their current systems.
  - **Supply:** the widespread nature and acceptance of cloud computing result in a wide supply of services.

- **Economic benefits.** Cloud computing has numerous advantages as regards IT project economic analyses and profitability studies (Forbes, 2011; Forrester Research, 2011).
  
  - **Contracting on demand:** the customer pays only for the use that s/he is making of the service. If that demand increases, then the technical capability of the cloud can be increased, while the load can be reduced during periods of less productivity, thus minimising costs.
• Dynamic supply: the supply of resources can be automated without the need for either manual intervention or a greater investment in infrastructures.

• Human resources: the contracting of personnel to manage the IT is the provider’s responsibility, thus enabling the customer to reduce his/her workforce.

• Physical space: no physical space is required for the location of the data processing centre, nor is it necessary to invest in equipping rooms (air-conditioning, fire extinguishers, etc.).

• Energy supply: savings can be made on the consumption of electricity in the installations. Environmentally-friendly tendencies may additionally suppose an increase in an organisation’s reputation.

• Competition among providers: the growth of supply is converted into a reduction in prices as the result of competition.

• Benefits related to use and operation: the cloud computing service customer delegates numerous management and maintenance tasks to the provider’s environment.

  • Maintenance and administration: the maintenance and administration tasks become the service provider’s responsibility.

  • Accessibility: resources can be accessed from any device that is connected to the network.

  • Shared resources: the possibility of sharing resources via the Internet permits the concurrence of users or the sharing of documentation.

  • Updating of applications: the management of the updating of applications or database systems can be carried out with much less impact on the service provided.

  • Ease with which to dispense with the service: in the case that the cloud computing service contracted is no longer required, it is possible for the customer to dispense with it without having to worry about the investments made.

1.2.4. Cloud Computing Forecasts and Market Estimates

All markets for public cloud services are continuing to undergo a high rates of growth, and this is demonstrated by various statistics (Columbus, 2016):

• The figure 1-1 shows the size of the hosting and cloud computing market from 2011 to 2019 (Statista Inc, 2016).
TBR predicts (see Figure 1-2) that worldwide public cloud revenue will increase from $80B in 2015 to $167B in 2020. (TBR, 2016)

IDC predicts cloud IT infrastructure spending will grow at a CAGR of 15.1% from 2014 to 2019, reaching $53.1B billion by 2019 (see Figure 1-3). By 2019, IDC predicts that cloud IT infrastructure spending will be 46% of the total expenditures on enterprise IT infrastructure. (IDC, 2016)
In 2015, Amazon Web Services (AWS) generated $7.88B in revenue with Q4 2015, up by 69% from the previous year. (VentureBeat, 2016)

Morgan Stanley predict that the revenue of Microsoft cloud products will have increased by 30% in 2018 (see Figure 1-4). Office365, CRM, and Azure will increase from 11% of their revenue in 2015 to 30% in 2018. (Morgan Stanley, 2016)
1.3. **Security in Cloud Computing**

Information Security is a critical factor to consider when implementing cloud computing services. Security-related questions are highlighted by both experts in the subject and potential users, since the customers’ confidence in this type of solutions has decreased and may limit its adoption. In order to stress the importance of security for cloud computing services, this section provides a summary of the principal concepts of security and the risks that threaten this new paradigm.

1.3.1. **Introduction to Information Security**

Information has become critical for all types of organisations as regards their business activities, and this has increased the need to manage it carefully. Information is just as important a resource as capital or work, it has a high strategic value and is considered to be an asset of the upmost importance, and should therefore be adequately protected (Dhillon, 2000). This situation is even more pressing in new generation and ICT-based organisations in which information is part of their business nucleus.

Corporate security is a classical term that reflects the efforts made by organisations to avoid business risks, thus allowing them to confront the various threats that may put their survival in danger. The traditional concept of security has been broadened to include the aforementioned information activities, giving place to a combination denominated as Information Security. Information and security have thus become linked and related aspects, as is shown by the fact that the information in any organisation is only as good as the security mechanisms that protect it. Information that is not reliable, owing to erroneous security policies, generates uncertainty and a lack of confidence in all business areas, thus having a negative impact on the organisation. Secure information is, meanwhile, a sign of certainty that contributes to generating value for the company.

Information Security is defined by the ISO standard organisation as being the protection of the confidentiality, integrity and availability of information; this may also involve other properties such as authenticity, responsibility, non-repudiation and trustworthiness (ISO/IEC 27001, 2013). A similar definition is provided by the “Information Systems Analysis and Risk Management Methodology” (MAGERIT - Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información), according to which it is the capacity of networks or information systems to resist, with a determined level of confidence, accidents or illicit or malicious actions that compromise the availability, authenticity, integrity and confidentiality of the data stored or transmitted and of the services that those networks supply or make accessible (MAP, 2005). These definitions can be used as a basis to conclude that the nucleus of Information Security consists of the following four characteristics:

- **Authenticity**: this permits operations to be carried out with the guaranteed confidence that the person using the information is really who they say they are.

- **Confidentiality**: This is understood in the sense that only authorised users can access the information, thus avoiding its diffusion among those users who do not have appropriate permission.
• Availability: this refers to the capability to access the information when it is needed, thus guaranteeing that the services supplied can be used when they are required.

• Integrity: this is the quality that guarantees that the information has not been modified by third parties, thus ensuring the completeness of the data and that it is correct.

In summary, Information Security is the function that is in charge of protecting information from a wide range of threats such that it is possible to ensure the continuation of the business whilst minimising risks and maximising the return on investment and opportunities. These objectives are attained via the implementation of adequate controls, including policies, processes, procedures, organisational structures and competent hardware and software.

1.3.2. Risks Associated with Cloud Computing

The adoption of cloud computing-based services is associated with new risks, owing fundamentally to the new technologies employed and the new organisational processes in relation to customers and suppliers. It is therefore necessary to identify what these risks are, thus enabling the appropriate decisions to be made as regards their management and minimisation in the implementation strategies of these services.

These risks have been identified by various organisms, some of which are the European Union Agency for Network and Information Security (ENISA), the various work groups in charge of maintaining the set of 27000 regulations, the Cloud Security Alliance (CSA), etc.

It is important to highlight the report periodically published by the CSA, which contains the greatest risks detected, compiled by means of surveys of professionals in this subject (Cloud Security Alliance, 2016b). According to these experts, the greatest risks confronted by cloud computing service customers are:

• Data Breaches: A data breach is an incident in which sensitive, protected or confidential information is released, viewed, stolen or used by an individual who is not authorised to do so.

• Insufficient Identity, Credential and Access Management: Data breaches and the enabling of attacks can occur because of a lack of scalable identity access management systems, failure to use multifactor authentication, weak password use, and a lack of ongoing automated rotation of cryptographic keys, passwords and certificates.

• Insecure Interfaces and APIs: Cloud computing providers expose a set of software user interfaces (UIs) or application programming interfaces (APIs) that customers use to manage and interact with cloud services. The security and availability of general cloud services is dependent on the security of these basic APIs.

• System Vulnerabilities: System vulnerabilities are exploitable bugs in programmes that attackers can use to infiltrate a computer system for the purpose of stealing data, taking control of the system or disrupting service operations.

• Account Hijacking: Account or service hijacking is not new. Attack methods such as phishing, fraud and exploitation of software vulnerabilities still achieve results
1.4 Legacy Systems Migration to Cloud

When a new technology emerges, the products that were developed or deployed with the preceding technologies can become legacy. Legacy is an ambiguous adjective: for the business management it means a valuable asset, for the IT department it means a risk.

1.4.1. Legacy Systems

Most companies have existing information systems that are considered to be legacy systems because the code in these systems was written long ago and may now be technologically obsolete. According to (Paradauskas & Laurikaitis, 2006), "a legacy information system is any information system that significantly resists modification and evolution to meet new and constantly changing business requirements". The authors in (Ulrich, 2002) state that "legacy systems are defined as any production-enabled software, regardless of the platform it runs on, language it written in, or length of time it has been in production", while those in (Hunt & Thomas, 2002) go further and state that the "code becomes legacy code just about as soon as it is written".
Software vendors have cultivated a belief that "anything new is beautiful and that everything old is ugly" and we have become "victims of a volatile IT industry" (Sneed, 2008). Despite the fact that legacy systems may be obsolete, this kind of system usually has a critical mission within the company and represents a valuable asset for companies, since legacy systems embed a lot of business logic and business rules that are not present elsewhere (Sommerville, 2004). The business knowledge embedded in legacy systems results from the fact that companies maintain their legacy systems over time. This maintenance adds increasingly more functionalities to legacy systems, supporting the company's operation and activities. Companies cannot therefore discard their legacy systems, although it is for this reason that they must deal with the underlying problems of software erosion in the legacy systems (Paradauskas & Laurikaitis, 2006):

- Legacy systems are typically implemented with obsolete technology which is difficult and expensive to maintain.
- The lack of documentation leads to a lack of understanding of the legacy systems, making the maintenance of these systems a slow and expensive process.
- A great effort must be made to integrate a legacy system into other systems in the company, since the interfaces and boundaries of the legacy system are not usually well defined.

1.4.2. Legacy Systems Migration

Reverse engineering techniques have become very important within the legacy system migration process. Firstly, reverse engineering makes it possible to retrieve abstract representations in order to facilitate the comprehension of different legacy systems. Secondly, abstract representations obtained by reverse engineering from legacy systems can be refactored to improve their maintainability or add new functionalities so as to evolve legacy systems.

Software modernisation is a specific new kind of evolutionary maintenance paradigm that is used to solve certain reengineering problems.

Business process archaeology has emerged as a set of techniques and tools with which to recover business processes from source code, that is, at a higher abstraction level. The maintenance benefits of business process archaeology are that it preserves business behaviour buried in legacy source code and retrieves business processes, thus providing more opportunities for re-factoring (owing to the higher abstraction level).

1.4.2.1. Reengineering

Two decades ago, the reengineering of legacy systems became one of the most successful practices as regards dealing with software erosion problems in this type of system. A reengineering process does not discard the whole system, and the enhanced system that it obtains preserves most of the business knowledge embedded in the legacy system. The reengineering processes therefore make it possible to carry out evolutionary maintenance of the legacy systems, assuming low risks and low costs (Sneed, 2005).
The reengineering process consists of three stages: reverse engineering, restructuring and forward engineering (Chikofsky & Cross, 1990):

- **Reverse engineering**: the first stage is the reverse engineering process which analyses the legacy system in order to identify the components of the system and their interrelationships. The reverse engineering stage then builds one or more representations of the legacy system at a higher level of abstraction.

- **Restructuring**: the second stage is the restructuring process which takes the previous system's representation and transforms it into another at the same abstraction level. This stage preserves the external behaviour of the legacy system.

- **Forward engineering**: finally, the third stage of the reengineering process is forward engineering, which generates physical implementations of the target system at a low abstraction level from the previously restructured representation of the system.

According to (Kazman, Woods, & Carrière, 1998), the whole reengineering process can be viewed as a horseshoe model (see figure 1-5). The left-hand side of the horseshoe represents the reverse engineering stage that increases the abstraction level; the restructuring stage preserves the same abstraction level, and is thus represented by the curve of the horseshoe; and the last part, on the right-hand side of the horseshoe, is the forward engineering stage that instantiates the target system.

![Figure 1-5. Horseshoe model for legacy software migration](image)

**1.4.2.2. Software Modernization**

The software modernisation paradigm, and particularly ADM (Architecture Driven Modernisation) as defined by the OMG (Object Management Group), can be considered as a mechanism for software evolution, i.e. it makes it possible to modernise the legacy information systems and eradicates, or at least minimises, the software erosion problem in legacy systems. According to (Force, 2004), “ADM is the process of understanding and evolving existing software assets. ADM restores the value of existing applications”.

ADM solves the problems of traditional reengineering since it carries out reengineering processes by taking model-driven principles into account. However, ADM does not replace reengineering, but rather improves it. The MDA (Model-Driven Architecture) standard proposed
by OMG defines two main principles: (i) modelling all the artefacts as models at different abstraction levels; and (ii) establishing model transformations between them (Miller & Mukerji, 2003).

As part of the effort undertaken by the OMG ADM Task Force, the KDM (Knowledge Discovery Metamodel) has been defined. Despite the fact that the KDM was defined by the OMG, it is being adopted as the international standard ISO/IEC 19506 ADM KDM (ISO/IEC, 2012).

The KDM is a specification of agreed upon facts and how these facts are represented in XML by using OMG standards such as Model Object Facility (MOF) and XML Metadata Interchange (XMI). The availability of this information represented according to the KDM specification makes it possible to (i) store the facts about information systems in a compliant structure; (ii) analyse and reason about KDM facts; (iii) exchange KDM facts for XML documents; and (iv) build tools so as to parse source code in a given programming language in order to generate language-independent and vendor-neutral facts about information systems.

From the modernisation perspective, KDM allows all the stages of modernisation processes based on ADM to be addressed: reverse engineering, restructuring and forward engineering.

1.4.2.3. **Business process archaeology**

Business process archaeology (Ricardo Pérez-Castillo, de Guzmán, & Piattini, 2011) studies the business processes in an organisation by analysing the existing software artefacts. The objective is to discover the business forces that motivated the construction of the enterprise information systems.

On the one hand, traditional archaeologists investigate several artefacts and situations in an attempt to understand what they are looking at, i.e. they must understand the cultural and civilising forces that produced those artefacts. Similarly, a business process archaeologist analyses different legacy artefacts, such as source code, databases and user interfaces and the entries, in order to learn what the organisation was thinking so as to understand why the organisation developed the information system in a particular way.

The business process archaeology initiative is being progressively supported by new reverse engineering techniques and tools with which to retrieve and elicit the embedded business knowledge.
1.5. **Hypothesis and Goals**

The hypothesis of our work is:

*The migration of legacy systems to secure cloud computing can be benefited by methodological frameworks*

Bearing in mind this hypothesis, the goal of our work can therefore be defined as:

*The definition of a methodological framework to support migration of legacy systems to secure cloud computing*

This main goal was used as a basis to define a set of partial objectives:

1. Analyse the existing proposals related to the frameworks, processes and methodologies that allow the security migration of legacy systems to cloud computing.
2. Study the different techniques and tools that support reengineering, specifically those that support software modernisation and business process archaeology.
3. Study the specific security proposals used to analyse security requirements and needs from the first stages of the software lifecycle.
4. Define the activities, tasks, steps, actors and artefacts needed to design an engineering process for the security migration of legacy systems to cloud computing.
5. Ensure the alignment of the process with international security standards and ensure its compliance with them.
6. Design and implement a tool prototype in order to provide the proposal with support.
7. Validate the proposal through its practical application in real scenarios.

1.6. **PhD Thesis Framework**

In this section, the framework in which the PhD thesis has been developed is presented. We shall first comment on the organisations at which the author has developed this work, after which the R&D projects that form the context in which this PhD thesis has been carried out are presented.
1.6.1. **Research Group**

The author of this PhD thesis carried out this work as a member of the GSyA R&D Group and also collaborates with the SenSe Group. His works have been oriented towards the research into security architectures, and specifically that into security for cloud computing.

The main goal of the Security and Audit Group (GSyA, http://gsya.esi.uclm.es) is to carry out research into the various issues related to the security and auditing of information systems, thus contributing to the improvement of teaching in computer engineering and providing solutions for the industrial sector. The GSyA Research Group is formed of professors from the Department of Technologies and Information Systems at the University of Castilla-La Mancha and is located at the “Escuela Superior de Informática in Ciudad Real”, where it has an R&D Laboratory.

The INCIBE has led the creation of the “Red de Excelencia Nacional de Investigación en Ciberseguridad” (National Excellence Network of Cyber-Security Research) of the 1st July, 2016, of which the UCLM is an associate founder, represented by the GSyA group.

The objective of the Secure and Dependable Software Systems Research Cluster (SenSe, http://sense-brighton.eu/) is to develop new and pragmatic ways in which to ensure the dependability of software systems with a particular emphasis on security, trust and risk. The experience of the SenSe cluster in the analysis and risk management is very useful for this thesis. The SenSe cluster is formed of professors from the University of Brighton.

1.6.2. **R&D Projects**

This work has been developed within the framework of the following research projects: SIGMA-CC, SERENIDAD y SEQUOIA. An overview of the most important aspects, in order of start date, of these projects is shown in the table 1-1.

<table>
<thead>
<tr>
<th>Project</th>
<th>Description</th>
<th>Main Reseacher</th>
<th>Period</th>
</tr>
</thead>
<tbody>
<tr>
<td>SERENIDAD (PEII-2014-045-P)</td>
<td>Advances in security and quality of the building of data warehouses based on models</td>
<td>Dr. D. Eduardo Fernández-Medina Patón</td>
<td>September 2014 – September 2016</td>
</tr>
<tr>
<td>SEQUOIA (TIN2015-63502-C3-1-R)</td>
<td>Security and quality in processes with big data and analytics</td>
<td>Dr. Dña. Marcela Genero Bocco and Dr. D. Eduardo Fernández-Medina Patón</td>
<td>January 2016-December 2018</td>
</tr>
</tbody>
</table>
A detailed description of the projects is shown below:

- **SIGMA-CC (TIN2012-36904):** The "Security Governance and Secure Systems Migration to the Cloud Computing" project aims to improve the way in which the information security is governed in cloud computing-based services, with special attention being paid to the strategy that applications and systems are migrated to the cloud, ensuring they are adapted to the new security challenges of this paradigm. It is funded by the Economy and Competitiveness Ministry. This PhD thesis is located within the main goals, with the definition of models, patterns and methodological frameworks for the information systems migration to the cloud computing, with particular attention being paid to new security challenges that arise in this paradigm.

- **SERENIDAD (PEII-2014-045-P):** Advances in security and quality of the building of data warehouses based on models. This has been funded by the "Consejería de Educación, Ciencia y Cultura" of the "Junta de Comunidades de Castilla La Mancha", and the "Fondo Europeo de Desarrollo Regional FEDER". The main goal of this project is to continue with the large number of open research lines concerning security and quality in the model driven approach used to build data warehouses, integrating security and quality aspects and indicators by using hybrid development paradigms, defining mechanisms that allow the traceability and validation of security and quality requirements throughout the life cycle, adding and adapting dynamic security models that add more security controls to data warehouse design, studying and applying security patterns, and adapting data warehouse security solutions to be integrated into the cloud. This PhD thesis is located within some of the objectives of this project, defining mechanisms with which to analyse the traceability of security requirements, and integrating the use of security standards into the design of cloud systems.

- **SEQUOIA (TIN2015-63502-C3-1-R):** Security and quality in processes with big data and analytics. It is funded by the Economy and Competitiveness Ministry. The main objective of the SEQUOIA project is to contribute models, methods and software tools with which to improve the quality and security in the context of Big Data. It involves a systematic and methodical way of seeing how to incorporate Big Data into the daily decision-making process, and it optimises the synergy between the different approaches of Business Engineering, Software Engineering and Data Engineering. The experience of this PhD thesis is serving as a starting point for some of the tasks in the project.

### 1.7. Document Outline

The subsequent chapters are structured as follows:

- **Chapter 2. Research Method.** This chapter presents the work method used to attain the stated goals.

- **Chapter 3: State of the Art.** In this chapter, a detailed study of the main areas related to the secure migration of legacy systems to cloud computing is carried out.

- **Chapter 4: Overview of the SMiLe2Cloud process.** This chapter provides a complete description of the SMiLe2Cloud process. It contains a description of the activities, tasks, actors, techniques and artefacts of which the SMiLe2Cloud process is composed.
• Chapter 5: Prototype. This chapter presents the prototype designed and implemented during the development of the PhD thesis. Its main objective is to show the capability of providing software based on a CASE tool that makes it easier for the developer to execute the SMiLe2Cloud process.

• Chapter 6: Case Study. In this chapter we present the application of the SMiLe2Cloud process described in Chapter 4 to a real application.

• Chapter 7: Conclusions. The final conclusions reached as a result of working on this thesis are presented in this chapter. The publications achieved, which guarantee the performance of the proposal, are also shown. Future research lines related to the SMiLe2Cloud process are additionally stated.

• Bibliography. List of bibliographical references mentioned throughout the text of this PhD thesis.

• Appendixes. The appendixes have been included in order extend and specify information for a better understanding of some of the aspects presented in previous chapters, such as acronyms.
2. Research Method
If research is to be rigorous and scientifically valid, then certain methodologies must be followed. In the particular case of this thesis, the research methods used to achieve the goals defined are the Action-Research method and a Systematic Literature Review. This chapter provides an introduction to both methods (Genero, Cruz-Lemus, & Piattini, 2014), along with how they have contributed to fulfilling the main objectives of this thesis.

2.1. **Action-Research Method**

Of all the qualitative research models found in literature (most of which originate from the area of social sciences), we have principally focused on that of Action-Research since it is the model which is most often used in Information Systems (IS) and Software Engineering (SE). The term “action-research” was coined by the author Kurt Lewin (Lewin, 1946), who used it to describe a research method that could link the experimental approach of social sciences to social action programs that provided solutions to the main social problems of that moment. Lewin stated that Action-Research makes it possible to achieve both theoretical advances and social changes simultaneously. This method has been widely accepted and has been applied to the research on Software Engineering since it was first used in 1985 by Wood-Harper (Wood-Harper, 1985).

In fact, Action-Research does not refer to a specific research method, but rather to a set of methods of the same type which share the following properties (Baskerville, 1999):

- Focus on action and change.
- Focus on a problem.
- An “organic” process model which involves systematic and interactive phases.
- Participants’ collaboration.

2.1.1. **Definitions**

Since it is not a specific method, there are many definitions of Action-Research, the most important of which are:

- That of (McTaggart, 1991), who stated that it is the way in which to meet the required conditions, to learn from our own experiences and make them accessible to others.
- That of (French & Bell, 1996), according to whom it is the process of collecting research data by means of systematic mechanisms. The data collected refers to a current system related to an objective or system requirement; feeding the system with that data; undertaking actions by means of alternative variables selected from the system, based on the data and the hypotheses; and evaluating the results of the actions by collecting additional data.
- That of (Wadsworth, 1998), who opined that it consists of the participation of all research members in studying the current problematic scenario in order to improve it or change it.

These definitions make it possible to deduce that Action-Research has two aims: to benefit the research “client” and to increase the research knowledge (Kock & Lau, 2001). Hence,
Action-Research is a collaborative type of research which seeks to make theory and practice meet, to establish a link between research and practice by means of a cyclical process. Action-Research is focused on yielding new knowledge which is useful in practice and which is gained by introducing changes and researching into candidate solutions to different real scenarios which are relevant to a group in practice (Avison, Lan, Myers, & Nielsen, 1999). This is achieved thanks to the intervention of a researcher in the real circumstances surrounding the group. The results of these experiences must be beneficial to both the researcher and the participants. A fundamental premise regarding this kind of research is the complexity of social processes (and the use of information technologies in this type of organisations), which can be better studied by making changes to those processes and observing the effects of the changes (Baskerville, 1999).

In the area of Information Systems, the client is generally an organisation that benefits from the researcher’s services. These services may be software development or maintenance consulting, which allow researchers to access data which is relevant to the research and to receive financing (Kock & Lau, 2001). However, those researchers that might use Action-Research in Software Engineering (SE AR) will always be serving two masters: The client and the Software Engineering scientific community. Their needs are usually quite different and sometimes opposed. Attempting to satisfy both of them is the main challenge that all SE AR researchers have to confront. This, however, brings with it a number of new elements to the research which adds to its interest.

2.1.2. Roles and Variations in the Action-Research

Since its origins, various means of applying Action-Research methodology have been developed (Chein, Cook, & Hardin, 1948). French and Bell (French & Bell, 1996) propose four variations, which basically depend on the characteristics of the research project:

- **Diagnosis**: The researcher comes up against a difficult situation; s/he diagnoses it and gives recommendations to the critical reference group, but without afterwards controlling the effects.
- **Collaborative**: The critical reference group puts in place the recommendations made by the researcher, and informs him or her of the results and effects.
- **Empirical**: The critical reference group carries out broad and systematic research into the situations and effects. This characteristic makes this variant difficult to implement.
- **Experimental**: This consists of evaluating the different options that exist in order to achieve an objective. The main disadvantage is that the different options are difficult to measure since they will generally be applied either in different organisations (with different characteristics, which may cloud the research results) or in one organisation but at different moments (signifying that the work environment could have changed).

In a more formal analysis of the participants in Action-Research, Wadsworth (Wadsworth, 1998) identifies the following four kinds of roles in this method (the same person or team may sometimes play more than one role):

- **Researcher**: The person or group of people who actively carry out the research process.
- **Researched object**: The problem to solve.
• Critical reference group: A group on which research is performed inasmuch as it has a problem.

• The problem that needs to be solved: Additionally, this group takes part in the research process but not as actively as the researcher. The critical reference group contains some people who know that they are participating in the research and others who do not.

• The stakeholder: Anyone that can benefit from the research but does not directly participate in it. Stakeholders may include organisations that are using a new method to solve IT problems or experts who apply those methods.

2.1.3. Stages of Action-Research

A research process which uses Action-Research consists of a set of activities that make up a characteristic cycle. Padak and Padak (Padak & Padak, 1994) identify four steps, which have to be followed in research when using this method.

1- Planning: This identifies the main issues which will guide the research and which should be directly related to the object of study and be solvable. This activity consists of identifying new alternative means of research, lines to follow, or the reinforcement of something. The result is that other problems or issues to be solved are clearly identified. Some authors (Baskerville, 1997) distinguish between diagnosis (identifying initial problems) and planning (specifying actions by which to solve those problems).

2- Action: Controlled, careful and deliberate variation of the practice. A simulation or test of the situation is carried out.

3- Observation: Collecting information and data, documenting what happens. This information may originate from different sources (literature, measures, results of the tests, interviews, etc.). It is also known as evaluation.

4- Reflection: Sharing and analysing the results with the relevant people, putting new interesting questions up for discussion “ (Wadsworth, 1998). Carefully studying the area that is being researched in order to yield new knowledge that could involve improvements, by making changes as part of the research process and studying what happens after those changes. This stage is also known as “learning specification”. In some Action-Research variants, it is not really a single stage, but rather an on-going process.

Given these characteristics, the process defined as Action-Research could be said to be iterative inasmuch as it moves forward to solutions which are more refined in a cyclical manner. In each cycle, new ideas are proposed which are tested in the next cycle, as shown in Figure 2-1. This cycle identifies Action-Research as being a reflective learning process and research into solutions. Action-Research is therefore cyclical, which means that actions are evaluated and re-planned so as to follow a diagnosis and reflection.
2.1.4. **Problems of Action-Research and Alternatives**

In recent years, Action-Research has been identified as one of the qualitative research methods most frequently used in the area of Information Systems and Software Engineering research. However, the community of experts has detected some problems in its application, for the following reasons:

- Lack of SE AR methodology.
- The consulting framework imposes an over-restrictive perspective since it implies contractual liabilities and organisational interests that could be detrimental to the research.
- The lack of a defined research process model which indicates the steps to follow in SE AR.

All the issues indicated above may be understood to imply that this is not a rigorous research process. In order to solve these problems, the following alternatives are proposed:

- The research should be carried out by using a project management perspective;
- The inclusion of quality criteria which are specially defined;
- An analysis of the factors that lead to the formalisation of a process; and
- The process should be organised by means of a project structure.

Estay and Pastor (Estay & Pastor, 2000a, 2000b) have combined some of these ideas to suggest using project management in order to improve the Information Systems (and SE) AR project formality. This means that a project structure which includes the main Information Systems AR elements should be used. In order to achieve the above goal, these authors indicate the need to adopt management practices which are suitable for Information Systems AR based
on the PMBOK (Guide to the Project Management Body of Knowledge) (PMI, 2000), an internationally recognised management model. According to Estay and Pastor, the concepts of Action-Research and Project are equivalent: they are both unique work experiences; their final results are also unique, and they share the notion of intervention, which means that both of them involve a voluntary change in reality. Action-Research intervention leads to changes in the work practice, but it is also a means to attain hands on experience data which are needed for the research process. The same authors have proposed a capability model based on the Capability Maturity Model (CMM) (SEI, 1995) which applies incremental management practices with the aim of guaranteeing rigour and quality improvement with regard to the use of Action-Research in Information Systems (Estay & Pastor, 2001).

2.1.5. **Action-Research Cycles**

In this Information Systems (and SE) qualitative research, it can be considered that there are two spheres (scientific and academic) which interact with each other in spite of their moving along different paths. IS/SE AR has a dual aim and is based on two Action-Research cycle types for two types of projects:

- **Cycles aimed at solving problems within the Information Systems (and SE) projects.** These projects consist of the development of computing solutions (be they computing, software development, information systems implementation or maintenance projects, etc.). In this case, the researcher is in charge of solving a problem.

- **Cycles focused on research.** These projects are related efforts whose intention is to seek a result. In this case, Action-Research offers a work methodology and a justification through which to get closer to a specific reality in order to test a theory or hypothesis.

The IS (and SE) AR project structure proposed by Estay and Pastor (Estay & Pastor, 2000b) defines two specific cycles:

- **A cycle aimed at creating a solution which provides new useful knowledge for practitioners and helps them to improve their practices.** Researchers connect to reality by means of an intervention. Research is used to build models and theories influenced by reality, and to gain knowledge. This research cycle is focused on solving problems.

- **A cycle focused on managing research in order to provide the Information Systems (and SE) subject with new knowledge and to improve the researchers’ practices.** This research interest cycle is focused on the research interest.

In short, IS (SE) AR can be analysed in two complementary manners (figure 2-2).

- **Vertically, based on the type of project.**

- **Horizontally, based on the typical bi-cyclic structure of an IS AR project.**
Lau (Lau, 1997) provides an outline of the use of IS/SE AR, including several examples published by different authors regarding the analysis, design and development of Information Systems, and particularly software implementation and related processes.

Baskerville (Baskerville, 1999) provides an introduction to the use of IS/SE AR, indicating ten IS (SE) Action-Research forms and four characteristics which determine the way in which Action-Research is used. These are as follows: Process Model (iterative, reflective, linear); structure (rigorous, fluid); typical involvement (collaborative, facilitative, expert); and primary goals (organisational development, system design, scientific knowledge, training).

Baskerville and Wood-Harper (Baskerville & Wood-Harper, 1996) list seven basic strategies that can be employed to achieve IS/SE AR: using the “change paradigm”, establishing an agreement or formal research contract, providing a theoretical framework, planning data collection methods, maintaining collaboration and mutual learning between the researcher and the critical reference group, providing incentives for the performance of the typical cycle interactions and seeking the generalisation of solutions.

2.1.6. Application of Action-Research to this PhD Thesis

Bearing in mind that the research object of this PhD Thesis is to “Define a methodological framework to enable the security migration of legacy systems to Cloud computing”, along with the fact that it is being developed within the framework of several R&D projects in collaboration with diverse Universities and Information Society Technologies (an external organisation), the application of the Action-Research qualitative method has been considered. We have decided to use the Action-Research collaborative variation because we consider it to be the most appropriate for the definition of the process.
To do so, we have taken into consideration the following participants:

- **Researcher**: the GSyA (Security and Audit) research group formed of professors from the School of Computer Science at the University of Castilla La Mancha in Ciudad Real, and the SenSe (Secure and Dependable Software Systems) research group, formed of professors from the University of Brighton. The author of this work is a member of the GSyA Research Group and collaborates with SenSe.

- **Researched Object**: the security migration of legacy systems to cloud computing.

- **Critical Reference Group (CRG)**: the proposals made in this thesis have been validated with the support of the Spanish National Authority for Markets and Competition (CNMC), and it has been possible to implement and validate the SMiLe2Cloud methodology in the sphere of this organisation. Moreover, the research is contained within the framework of the SIGMA-CC, SERENIDAD, SEQUOIA projects, on which various Spanish and European Universities collaborate. The critical reference group is, therefore, composed of representatives from the above mentioned projects, along with those from Spanish and European universities.

- **Stakeholders**: these will be, on the one hand, all those organisations that could potentially benefit from the results of the work; in other words, all software development and maintenance enterprises that develop software systems in which the security aspect is considered critical. More specifically, the CNMC to which the methodology has been applied and validated has attained enormous benefits from the adoption of the SMiLe2Cloud process originating from this thesis, since the task of integrating the security aspect into the migration or REM application to cloud computing has been guaranteed, normalised and facilitated. The scientific community will, on the other hand, reuse the results to carry out further research in these fields.

Figure 2-3 shows the application of the collaboration-action variation to our work.

The application of SE-AR is more obvious when a human organisation interacts with information systems. In fact, Action-Research is one of the few approaches that is valid as regards studying the effects that specific alterations have on systems development and maintenance methodologies in human organisations (Baskerville & Wood-Harper, 1996).
The beginning of Action-Research during the research process employed in this work implied feedback between the researcher and the CRG. This enabled us to identify three groups of basic cycles involved in the application of Action-Research to our development process, which consequently allowed us to obtain successively more refined solutions generated in a collaborative manner. These solutions were focused on the definition of the sub-processes of the proposed development process, including their tasks, actors, inputs, outputs and key security artefacts. This process can be summarised in the following cycles:

- **Initial general cycle:** The researchers, together with the critical reference group, defined the general problem of the security migration of legacy systems to cloud computing. They also established the objectives and general requirements of the process needed (planning). They then proceeded to search for all information that might be of interest (action). Its further analysis (observation) allowed them to discover that the object of study was very complex, given that many different kinds of aspects had to be taken into account. The reasoning and sharing (reflection) between researchers and the CRG made it possible to detect that the solution might consist of approaching all the stages of the migration of legacy systems to cloud computing in an integrated manner. In summary, this cycle consisted of studying the different aspects influencing the migration of legacy systems to cloud computing. As a result of this general cycle, we identified the two main research cycles (conceptual and technical).
2.2 Systematic literature review

Intermediate general cycles: The intermediate cyclic groups identified in the general cycle were the following:

- Conceptual cycle: What is it necessary to define in a methodological framework that will allow the security migration of legacy systems to cloud computing?
- Technical Cycle: What tools, standards and software technologies are useful as regards defining, in an integrated manner, the modelling of the elements related to a framework that will allow the security migration of legacy systems to cloud computing?

Final specific cycles: From the moment at which the previous answers were clarified by both researchers and the critical reference group, we proceeded to carry out specific Action-Research cycles for each of the two main components.

The aforementioned cycles implied that for the development of the proposal on which this PhD thesis is based it was necessary to use Action-Research with a multi-cyclic project structure.

The results of the application of this work method, which was employed to attain the objectives stated in the thesis, are described in detail in Chapters 4, 5, and 6.

2.2. Systematic Literature Reviews

Almost all research starts with some kind of literature review. From a scientific perspective, the literature review should be complete, systematic and replicable in order to find relevant studies related to the phenomenon being studied (B. Kitchenham & Charters, 2007). Systematic Literature Reviews (SLRs) and Systematic Mapping Studies (SMSs) can be performed with this purpose. An SLR is described as a form of secondary study that uses a well-defined methodology to “identify, evaluate and interpret all available evidence relevant to a particular research question, topic area, or phenomenon of interest” (B. Kitchenham & Charters, 2007). An SMS is, meanwhile, described as being a study that provides a structure of the type of research reports and results that have been published by categorising them (Petersen, Feldt, Mujtaba, & Mattsson, 2008).

Both kinds of studies can be carried out by using the same procedures to select primary studies (Wohlin et al., 2013). The method used to conduct an SLR is, therefore, described in Section 2.3.1., while the features of the SMS are shown in Sections 2.3.2.

2.2.1. Method Used to Conduct a Systematic Literature Review

Kitchenham proposed a method with which to conduct systematic literature reviews (B. Kitchenham, 2004; B. Kitchenham & Charters, 2007), which was based on the guidelines developed for medical research and was later adapted for use in the sphere of Software Engineering. The reasons why this approach is so useful can be summarised as follows:

- It provides a summary of the existing evidence concerning a particular sphere or technology.
- It identifies lacks in the state of current research, thus suggesting areas in which improvements can be made in the future.
It provides a framework in which to appropriately locate new research activities.

A systematic literature review is defined as a means to evaluate and interpret all available research related to a particular research question, as regards a thematic area or phenomenon of interest (B. Kitchenham, 2004). The individual studies in this sphere that contribute towards a systematic review are denominated as primary studies. A systematic review therefore constitutes a form of secondary study, since it is based on previous studies.

The proposed method consists of three fundamental stages, which are in turn divided into various activities (B. Kitchenham & Charters, 2007):

- Planning the review
  - Identifying the need for a review
  - Commissioning the review
  - Specifying the research question
  - Defining the review protocol
  - Evaluating the review protocol
- Developing the review
  - Identifying research
  - Selecting primary studies
  - Evaluating the quality of those studies
  - Extracting and monitoring data
  - Data synthesis
- Publishing the results of the review
  - Specifying diffusion mechanisms
  - Formatting the main report
  - Evaluating the report

The method shown can be adapted to the magnitude of the research in question. In the particular case of this doctoral thesis, some of the activities have been slightly simplified. The stages followed to conduct the systematic literature review used in this thesis are shown below.

2.2.1.1. Stage 1: Planning the review

The objective of the first stage is to define the most important parameters to be borne in mind when conducting the systematic review. It is necessary to establish the reasons and motives that justify conducting it, the way in which the search for works is carried out and how these will subsequently be reviewed. Finally, it is also necessary to include an evaluation of the planning itself.
Identifying the need for a review

The need to conduct a systematic review arises when it is necessary to gather together all existing information on a particular phenomenon in a rigorous, impartial and unbiased manner. This need may arise because it is necessary to obtain more general conclusions about the phenomenon that is the objective of the research than can be obtained by means of individual studies, or as a prelude to future research activities.

Specifying the research question

The specification of the research question is the most relevant activity in the systematic review process. This question directs and conditions the following stages of the review: the search process should identify primary studies that fit the question posed; the data extraction should revolve around information that will allow the question to be answered; and the analysis of the data extracted should make it possible to obtain response to the research question.

Defining the review protocol

The review protocol specifies the methods that will be used to develop the systematic literature review. This protocol is necessary to reduce the possibility of researcher bias. The review protocol usually consists of the research questions, the primary study search strategies, the study selection criteria and procedures (inclusion and exclusion), the criteria employed to evaluate the quality of the studies, the data extraction strategy, data analysis, the strategy employed to diffuse information and the time restriction of the project.

Evaluating the review protocol

The researchers should establish a procedure that can be used to evaluate the protocol defined. This evaluation should, if possible, be carried out by experts who are independent of the systematic review and who can, moreover, take charge of later reviewing its results. Nevertheless, in the case of reviews conducted in the sphere of doctoral theses, the protocol can be evaluated by tutors or those guiding the thesis.

2.2.1.2. Stage 2: Developing the review

Once the review protocol has been established, it is possible to begin the systematic review itself. The steps followed when developing a review are shown as follows.

Identifying research

The identification of the research consists of defining a search strategy, managing the primary studies and bibliography obtained and documenting the results of the search.
Selecting primary studies

Once the potentially relevant primary studies have been collected together, it is necessary to evaluate them in order to select those that are truly significant for the systematic review. This evaluation is carried out on the basis of selection criteria that make it possible to identify those studies that are directly related to the research question. It is at this stage that the inclusion and exclusion criteria defined during the review protocol are applied. It is also appropriate to carry out some sort of test to verify the suitability of the criteria defined, in case it is necessary to refine or modify them.

Evaluating the quality of the studies

In addition to the inclusion and exclusion criteria, it is also important to evaluate the quality of the primary studies selected so as to investigate possible differences in the results of the studies, interpret the discoveries made and consider the weight of their inferences. The objective of the evaluation of quality is to reduce any bias in the systematic review whilst simultaneously maximising its validity.

Extracting and monitoring data

The objective of this step is to design data-extraction forms that will enable the researchers to register the information obtained from the primary studies in a precise manner. In order to reduce the bias of the review, it is necessary to test the forms defined in the protocol so as to validate them before their application to all the studies. The extraction forms should include any aspect that might be necessary to answer the research question, in addition to standard information, such as the date of data extraction and publication details.

Data synthesis

Data synthesis consists of compiling and summarising the results obtained from the primary studies. This synthesis can be qualitative or quantitative, although it can on occasions also be both.

2.2.1.3. Stage 3: Publishing the results of the review

The last stage of the systematic literature review consists of reflecting on the results obtained after reviewing a paper or report and beginning to diffuse them in such a way that they will reach all possibly interested parties.

Specifying the diffusion mechanism

Given the importance of managing to efficiently communicate the results of the systematic review, it is recommended that a strategy for their diffusion be defined during the definition of the review protocol. This diffusion generally takes place via publication in journals or conferences related to the scientific sphere of which the review is part, but other review strategies can also be considered depending on the objective desired. For example, if the
intention is to influence particular professionals, it is possible to publish in specialised journals, the general press, on websites or to communicate directly with the interested parties.

**Formatting the main report**

The results of the systematic review are normally reflected in at least the following formats:

- In a technical report or in a section of a doctoral thesis.
- In a paper published in a scientific journal or at a conference.

In the case of a paper published in a scientific journal or at a conference, there tend to be greater restrictions as regards its length, which may make it necessary to refer to another longer report in order to show the rigor and validity of the systematic review. The format used in the report to reflect the systematic review should be appropriately structured so as to guarantee its understanding.

**Evaluating the report**

The evaluation of the report of the systematic review will be carried out by both its creators and the public at whom it is directed. On the one hand, throughout the development of the review, the same experts who evaluated the protocol defined should be in charge of supervising the different stages and evaluating the final report. On the other, it should also be evaluated by the reviewers at the journal in which it will hopefully be published, tutors or those guiding the doctoral thesis, or researchers and professionals who consult technical reports or websites.

### 2.2.2. Systematic Mapping Studies

A systematic mapping study (SMS), also known as a scope study, is a type of secondary study similar to an SLR. It uses the same rigorous method to identify relevant studies that answer the research question(s).

The main difference between both literature review methods is, however, the type of research question used. While an SLR attempts to answer a very specific question related to competing technologies, an SMS attempts to identify and classify the reported research related to a broad topic (Budgen, Turner, Brereton, & Kitchenham, 2008). The main purpose of an SMS is to provide an overview of a topic area and to identify areas for further empirical research.

Other researchers have identified slight methodological differences that could be considered when a researcher performs an SMS. An SMS should include both empirical and non-empirical studies. In addition, an SMS would be performed before an SLR because the outcome of the SMS could show clusters of empirical studies that could be inputs for the SLR method (Petersen et al., 2008).

Since an SMS can be used to support further research activities, it should be of high quality. The citation and classification of all references, the use of a well-defined and robust taxonomy and the application of a stringent search process are expected (B. A. Kitchenham, Budgen, & Brereton, 2011). An SMS that meets quality criteria can be used to: establish a baseline with which to analyse trends over time, provide a rationale for further empirical
research, identify relevant literature in order to elaborate a related work section of other studies, validate the search for results in the SLR itself, and support educational ends (B. A. Kitchenham et al., 2011).

However, if an SMS does not meet the quality criteria, it can be used as starting point for a more detailed SMS.

2.2.3. **Application of Systematic Literature Reviews to this PhD Thesis**

A systematic literature review has been employed during the development of this doctoral thesis. This methodology has been used to establish a solid base as regards the existing works related to the object being researched, i.e. as regards the security migration of legacy systems to cloud computing. Given the cyclical and dynamic nature of the research, the systematic review has been applied throughout the entire development of the thesis, with particular emphasis being placed on the first stages.

It was considered more appropriate to conduct an SMS. This was done by using the same method as that employed for an SLR, but adapting it to the research that we intended to carry out, i.e. some of the steps have been simplified.

2.2.3.1. **Research question**

The research question, which is a critical aspect in the first stage of a systematic review, was defined so as to identify the state of the art in the sphere of this doctoral thesis, in order to define the basis that would be used to locate this research within current scientific knowledge. The research question was focused on identifying initiatives, proposals or frameworks regarding the security migration of legacy systems to cloud computing. This question covered two objectives:

- The review proposed to seek complete and exhaustive migration frameworks, i.e. proposals that cover any type of aspect concerning the migration of legacy systems to cloud computing.
- The proposals had to take into account the special characteristics of security, and only those initiatives that had been designed for this paradigm would, therefore, be considered.

The research question allowed us to obtain a group of terms or keywords for the search. These keywords have been used to identify all those primary studies that might provide a response to the research question, without missing anything relevant. Bearing in mind that the majority of the publications in this sphere are in English, this language was used almost exclusively during the systematic review. The terms used in the systematic review search were the following:

- Legacy Systems Migration
- Legacy Systems Migration to Cloud Computing
- Cloud Computing Security
2.2 Systematic literature review

- Security Migration of Legacy systems to Cloud Computing

2.2.3.2. Source Selection

The sources used to conduct the systematic review were a set of databases, some of which have their own search engines, and which were chosen owing to their prestige and publications in the scientific community. The review was, therefore, conducted in a wide variety of sources, including electronic databases, search engines, journals and conferences. The principal sources employed are shown as follows:

- ScienceDirect
- Elsevier
- Google Scholar
- IEEE
- ACM Digital Library
- The Universidad of Castilla-La Mancha Digital Library

2.2.3.3. Study Selection

The combination of the search terms applied to the proposed sources resulted in a large number of publications.

As an example, the Figure 2-4 shows the results obtained from Google Scholar when using the aforementioned terms:

![Figure 2-4. Publications in Google Scholar](image-url)
The Figure 2-5 shows the evolution over recent years. As will be noted in the figure, the interest in the various terms is constantly evolving (it should be stressed that the results for 2016 refer only to those obtained up to 15th June):

![Figure 2-5. Evolution of publication in Google Scholar](image)

The quantity of publications obtained signified that they had to be filtered in order to attain the relevant primary studies. This was done by defining various inclusion and exclusion criteria. These criteria took into account both formal aspects, such as considering papers published after 2006, and aspects regarding content, such as verifying that the proposal had been adapted to standards. The verification of these criteria was, in the majority of cases, carried out by studying the title and abstract of each publication. However, in those cases in which doubts arose it was necessary to study the complete text.

2.2.3.4. Data Extraction

Once the primary studies had been identified, we then went on to analyse and compare them. This analysis was facilitated by carrying out extraction and synthesis processes for the data in each proposal. The results of the analysis of the systematic literature review are principally located in the chapter on the State of the Art of this thesis.

2.2.3.5. Publishing the Results of the Review

The results obtained after analysing the systematic literature review were published in (Márquez-Alcañiz, Rosado, Mellado, & Fernández-Medina, 2014).

The most detailed study of the results of the systematic review can be found in Chapter 3, State of the Art.
3. State of the Art
3.1. Cloud Information Security System Proposals

This chapter contains a description of the most relevant proposals in two spheres: Cloud information systems security and the migration of legacy systems to the cloud. In both cases, the Systematic Literature Review method has been employed with the objective of obtaining their current State of the Art.

The chapter concludes with a comparison of the principal proposals identified with the aim of highlighting the strong points and lacks of each one in an understandable and intuitive manner. The results of this comparison make it easier to understand the gap in the knowledge upon which this doctoral thesis is focused.

3.1. Cloud Information Security System Proposals

This section presents the methodologies and proposals regarding Information Systems Security found in literature. First, the standard and reference documents are reviewed, after which the various proposals concerning security methodologies for the development of information systems are analysed. Finally, an analysis of the range of methodological cloud computing security proposals is carried out.

3.1.1. Standards and Reference Documents

The following standards and reference documents regarding information system security, and specifically security in cloud computing, have been selected. Each of them is presented below, along with a brief description of their scope and principal characteristics.

3.1.1.1. Cloud Security Alliance

The CSA (Cloud Security Alliance) is a non-profit organisation whose objective is to promote the use of better security practices in cloud computing services and to provide training in the secure use of these services. Its members include security professionals, corporations, associations and other relevant actors from industry. Since its creation in 2008, the CSA has carried out active research in the sphere of cloud computing service security, whose results have been published in diverse documents and proposals, the most relevant of which are described as follows.


The CSA's principal contribution in this sphere is the Security Guide to Critical Areas of Interest in cloud computing (Cloud Security Alliance, 2011), which is a compilation of the best security practices compiled by experts in this area. These practices are distributed throughout 14 domains which deal with aspects such as the security architectures of cloud computing services or their governance and operation. The guide attempts to reinforce aspects such as stability, privacy and information security in these types of environments, and constitutes a first
level of approximation for both executives or managers and users or security implementers. In order to emphasise its practical aspect, each domain includes recommendations and requirements that illustrate the possibilities as regards applying the concepts and proposals provided. The recommendations contained in the guide should be evaluated by each organisation in order to determine which are applicable in their case; i.e. their use is not obligatory, but they rather help to identify possible threats and decide the best way in which to minimise associated risks.

The domains included in this guide are shown in Figure 3-1:

![Figure 3-1. CSA Domains](image)

3.1.1.1.2. **CSA Security, Trust & Assurance Registry (STAR)**

CSA STAR is a programme for security assurance in the cloud. STAR encompasses key principles of transparency, rigorous auditing and the harmonisation of standards. STAR certification provides multiple benefits, including indications of best practices and the validation of the security position as regards that which is offered by the cloud.

STAR consists of three levels of assurance, which currently cover four unique offerings, all based upon a succinct yet comprehensive list of cloud-centric control objectives in the CSA’s Cloud Controls Matrix (CCM).

The STAR programme includes a complimentary registry that documents the security controls provided by popular cloud computing providers. This publicly accessible registry was
designed to enable users of cloud services to assess their cloud providers, security providers and advisory and assessment services firms in order to make the best procurement decisions.

CSA STAR is based upon two key research components:

- **Cloud Controls Matrix (CCM)** - As a control framework, the CSA CCM provides organisations with the necessary structure, detail and clarity relating to information security tailored to cloud computing. CCM is the only meta-framework of cloud-specific security controls, and is mapped onto leading standards, best practices and regulations (ISO/IEC 27001/27002/27017/27018, COBIT or NIST).

- **The Consensus Assessments Initiative Questionnaire (CAIQ)** - Based upon the CCM, the CAIQ provides a set of Yes/No questions a cloud consumer and cloud auditor may wish to ask of a cloud provider to ascertain their compliance with the Cloud Controls Matrix and CSA best practices.

### 3.1.1.2. **ISO/IEC Standards**

The objective of the ISO/IEC 27000 family of standards is to provide a framework for the establishment, implementation, operation, motorisation, review, maintenance and improvement of Information Security Management Systems (ISMSs). It is composed of a set of standards that organisations can use to develop and implement a security management system for their information assets.

The ISO/IEC 27001 standard (ISO/IEC 27001, 2013) specifies the requirements that the organisation should fulfil for the implementation and operation of the ISMS in the context of business risks. It has been designed to ensure the selection of an appropriate set of security measures that will protect the information assets and provide the interested parties with confidence.

In addition to the aforementioned requirements, the regulation also provides indications regarding the management of resources, security procedure documentation, security audits and the management of and improvement to security processes.

The ISO/IEC 27002 standard (ISO/IEC 27002, 2013) complements the ISO/IEC 27001 standard, and describes numerous security measures such that they can be used to manage an organisation’s security system. These measures are grouped into 39 safety objectives, which are classified in 11 areas, including the corresponding implementation guide for each one.


ISO/IEC 27017 (ISO/IEC 27017, 2015) provides guidelines for information security controls that are applicable to the provision and use of cloud services by providing additional implementation guidance for relevant controls specified in ISO/IEC 27002, and additional controls with implementation guidance that specifically relate to cloud services.

ISO/IEC 27018 (ISO/IEC 27018, 2014) establishes commonly accepted control objectives, controls and guidelines for the implementation of measures to protect Personally Identifiable Information (PII) for the public cloud computing environment. In particular, ISO/IEC 27018 (ISO/IEC 27018, 2014) specifies guidelines based on ISO/IEC 27002 (ISO/IEC 27002, 2013), taking into consideration the regulatory requirements for the protection of PII which might be
applicable within the context of the information security risk environment(s) of a provider of public cloud services.

The ISO/IEC 27036 standard (ISO/IEC 27036, 2014) is focused on the Information Security of relationships with providers. It principally offers guidance and support in the evaluation and minimisation of the security risks involved in the acquisition and use of information or ICT services provided by external providers. The principal security risks related to the externalisation of services dealt with in this standard include: the dependency of external providers, which complicates the continuance of business; the difficulty of trusting in externalisation; the shared responsibility as regards conforming to existing policies, contracts or regulations, or coordination with the provider in order to adapt to dynamic security requirements that change over time.

Other ISO/IEC standards related to cloud computing are ISO/IEC 17788 (ISO/IEC 17788, 2014) and ISO/IEC 17789 (ISO/IEC 17789, 2014). ISO/IEC 17788 (Cloud computing – Vocabulary and overview) defines key cloud terminology and provides an overview of cloud computing. ISO/IEC 17789 (Reference architecture) covers general concepts and characteristics of cloud computing, the components/functions and roles and their capabilities and interrelationships.

Also under development is the ISO/IEC 19086 standard (Service Level Agreement Guidance) that provides an overview of SLAs (Service Level Agreement) for cloud services.

3.1.1.3. ENISA

The European Union Agency for Network and Information Security (ENISA) is a centre of expertise for cyber security in Europe. The Agency works closely together with Member States and the private sector to deliver advice and solutions. This includes studies on secure cloud adoption.

It has developed various documents related to cloud security, such as the cloud security risk assessment (Catteddu & Hogben, 2009). As a follow up to this risk assessment, they published an assurance framework with which to govern information security risks when moving to the cloud.

More recently, it has continued working on the following activities related to the cloud: managing security through SLAs, critical cloud services, cloud security and resilience, good practice guide for governmental clouds, incident reporting for cloud computing, certification in the EU cloud strategy, cloud certification schemes list (CCSL), etc.

Some of the most recent publications from ENISA that are related to cloud security are a Cloud Security Guide for SMEs (Dekker & Liveri, 2015) and a Security Framework for Governmental Clouds.

3.1.1.4. NIST

The NIST (National Institute of Standards and Technology), which is a United States Department of Commerce agency, has published various items relating to cloud computing services in general and to its security management in particular. In fact, the definition of cloud
computing proposed by the NIST is that which is most widely accepted by the scientific and technical community (Mell & Grance, 2011).

The NIST has also published a reference architecture with the intention of complementing its own definition of cloud computing. It is consistent with that definition and is based on the specification of the actors and roles involved (Liu et al., 2011).

It has, for some time, been working on the creation of a cloud computing security reference architecture (NIST, 2013), which is currently in draft form.

3.1.1.5. CCN - National Cryptology Centre

One of the documents published by the Spanish National Cryptology Centre (Centro Criptológico Nacional (CCN)), in an attempt to fulfil its objective of diffusing ICT security guidelines and recommendations for the State General Administration, is a guide that specifically deals with security in cloud computing environments (Centro Criptológico Nacional, 2013). This guide forms part of the CCN-STIC family of regulations and contains recommendations with which to enable the control of security risks introduced by cloud computing services, such that both the existing legal frameworks (including the National Security Schema or data protection regulation) and the security requirements established by the organisation in each case are adapted.

The content of the guide is focused on identifying the security requirements of the customer organisation, such that it is possible to give the provider specific details, thus guaranteeing the security of the service provided. Despite being oriented towards Spanish legislation and regulations, it contains proposals that are relevant to cloud computing service security.

According to the CCN, the appropriate management of security in these services begins with a correct risk management. IT proposes that risks be identified by using an existing methodology, such as MAGERIT, but that it should be borne in mind that the cloud computing paradigm includes specific risks that may not be considered in the chosen methodology. This risk management will also include the identification of threats to and vulnerabilities of the service.

3.1.1.6. ISACA

The ISACA (Information Systems Audit and Control Association) has published various items regarding cloud computing service security. That which most stands out is the Guide to Control Objectives for Cloud Computing, which provides information about this new paradigm, identifying the risks and controls that can be implemented (ISACA, 2011). This guide covers recommendations for the governance of cloud services, aspects of security that should be taken into account and requirements that customers and providers should evaluate in order to ensure these services.

More recently, ISACA has published a guide for the control and assurance of the cloud using COBIT 5 (ISACA, 2014).
3.1.2. **Security Methodologies for the Development of Information Systems**

A Systematic Literature Review was carried out, as the result of which the following relevant studies were selected, all of which are related to the integration of security into the systematic secure information systems development process. Each of these studies is presented and a brief description of their reach and principal characteristics is also provided.

3.1.2.1. **Secure Tropos**

Secure Tropos (Mouratidis & Giorgini, 2007) is a security-oriented extension of Tropos, a goal-oriented requirements engineering method.

Tropos is a software development methodology that has been adapted to describe both the organisational environment of a system and the system itself. Tropos adapts the i* modelling framework (Yu, 1995) which uses the concepts of actors, goals, tasks, resources and social dependencies. Tropos defines the concept of an Actor as an entity that has strategic goals and intentions within the system or within an organisational setting. An actor can be human, a system or an organisation. A Goal represents an actors’ strategic interests. A Plan (also known as a task) represents, at an abstract level, a way of doing something. The execution of a plan can be means to satisfy a goal. A resource represents a physical or an informational entity. A dependency between two actors indicates that one actor depends, for some reason, on another in order to attain some goal, execute some plan, or deliver a resource.

Secure Tropos consists of a language and a process that is focused on the requirements engineering stage. The language employs concepts from the requirements, security and privacy engineering domains, and it is based on the work on security requirements engineering.

Secure Tropos includes the concept of security constraint, which is defined as a restriction related to security issues, such as privacy, integrity, and availability. Security constraints can influence the analysis and design of the information system under development by restricting some alternative design solutions, by connecting with some of the requirements of the system, or by refining some of the system’s objectives. In addition, Secure Tropos defines secure dependencies which introduce security constraints that must be fulfilled for the dependency to be satisfied. A security mechanism represents potential solutions for the implementation of the security constraints, leading to the fulfilment of security objectives. The advantages of this approach, when compared to other security-oriented software engineering approaches, are: i) its ability to perform social analysis during the early requirements stage, ii) the simultaneous consideration of security and the other requirements of the system-to-be, iii) the support for not only requirements stages but also design stages.

Secure Tropos has a methodological approach which focuses on security requirements engineering. Some efforts have recently been made to consider the cloud computing paradigm.
(Kalloniatis et al., 2014). It is supported by certain tools, such as the University of Brighton’s SecTro.

3.1.2.2. **Comprehensive, Lightweight Application Security Process (CLASP)**

Originally defined by Secure Software and later donated to the Open Web Application Security Project (OWASP), CLASP is a lightweight process for building secure software (OWASP, 2006). It includes a set of 24 top-level activities, which can be tailored to the development process in use.

Key characteristics include:

- **Security at the centre stage**: The primary goal of CLASP is to support the construction of software in which security plays a central role. Furthermore, the activities of CLASP are defined and conceived primarily from a security-theoretical perspective and, hence, the coverage of the set of activities is fairly broad.

- **Limited structure**: CLASP is defined as a set of independent activities that have to be integrated into the development process and its operating environment. The choice of the activities to be executed and the order of execution is left open for the sake of flexibility. Moreover, the execution frequency of activities is specified per individual activity and, hence, the coordination and synchronization of activities is not straightforward. Two road maps (‘legacy’ and ‘green-field’) have been defined to give some guidance on how to combine the activities in a coherent and ordered set.

- **Role-based**: CLASP defines the roles that can have an impact on the security position of the software product and assigns activities to these roles. Roles are responsible for the finalization and the quality of the results of an activity. As such, roles are used as an additional perspective to structure the set of activities.

- **Rich in resources**: CLASP provides an extensive set of security resources that facilitate and support the implementation of the activities. For instance, one of these resources is a list of 104 known security vulnerabilities in application source code (e.g., to be used as a checklist during code reviews).

CLASP has a methodological approach and considers security to be a fundamental element. However, it is not focused on either the cloud computing paradigm or the migration of legacy systems.

3.1.2.3. **Trustworthy Computing Security Development Lifecycle (SDL)**

As a result of its commitment to trustworthy computing proclaimed in 2002, Microsoft defined the SDL (Lipner, 2004) in order to address the security issues frequently confronted by their products. SDL comprises a set of activities, which complement Microsoft’s development process and which are particularly aimed at addressing security issues.

SDL can be characterised as follows:
- Security as a supporting quality: The primary goal of SDL is to increase the quality of functionality-driven software by improving its security position. Security activities are most often related to functionality-based construction activities. For instance, threat modelling starts from architectural dependencies with external systems, while an architecture could in fact reduce such threats in the first place. SDL is designed as an add-on to the software construction process.

- Well-defined process: The SDL process is well organised and its related activities are grouped in stages. Although these stages are security specific, it is straightforward to map them onto standard software development phases. Furthermore, several activities have a continuous characteristic in the SDL process, including threat modelling and education. As such, the SDL process incorporates support as regards revising and improving intermediate results.

- Good guidance: SDL does a good job at specifying the method that must be used to execute activities, which, on average, are concrete and often somewhat pragmatic. For instance, attack surface reduction is guided by a flow chart and threat modelling is described as a set of sub-processes. As a result, the execution of an activity is fairly easily achievable, even for less experienced people.

- Management perspective: SDL takes a management perspective for the elicitation and description of many activities. This is an attractive aspect, given the inherent complexity of security, and it shows that security as a quality has to be managed in order to be realised in practice.

SDL has a methodological approach and considers security to be a fundamental element. Risk analysis and management are very important in the methodology. However, it does not focus on the cloud computing paradigm or on the migration of legacy systems. The methodology is supported by various tools.

### 3.1.2.4. Touchpoints

Touchpoints (McGraw, 2006) provides a set of best practices that have been distilled over the years thanks to its proposer’s extensive industrial experience. Most of the best practices, from here on denominated as activities, are grouped together in seven touch points.

Touchpoints can be characterised as follows:

- **Risk Management**: Touchpoints acknowledges the importance of risk management when it comes to software security. It tries to bridge the gap by elaborating a Risk Management Framework (RMF) that supports the Touchpoints activities.

- **Black vs. White**: The touch points provide a mix of black-hat and white-hat activities, both of which are necessary if effective results are to be attained. Black-hat activities concern attacks, exploits and breaking software (e.g., penetration testing). White-hat activities are more constructive in nature and cover design, controls and functionality (e.g., code review).

- **Flexibility**: The touch points can be tailored to the software development process already in use. To facilitate this, the documentation provides a prioritization of the
different touch points. This allows companies to gradually introduce the touch points, starting from the most important.

- **Examples**: Touchpoints is rich in examples. For instance, when describing abuse cases, there is an example giving the reader a good idea about what they might look like in a particular situation.

- **Resources**: To further aid the execution of activities, Touchpoints provides links to resources and also explains how to use them. To this aim, a part of the book is dedicated to security knowledge (which the resources are part of). For instance, attack patterns are provided in order to be used in the elicitation of abuse cases.

Touchpoints is a set of best practices focused on the construction of secure information systems. It is not especially dedicated to either the cloud or the migration of legacy systems.

### 3.1.2.5. UMLSEC

This proposal (Jurjens, 2002; Jürjens, 2005; Jurjens & Doser, 2005) extends the Unified Modelling Language (UML) by incorporating the security properties model into the information system. The specification of the requirements modelled with this security language is focused on access policies, confidentiality and integrity. This proposal is additionally supported by tools that allow the user to verify whether the designs generated fulfil the security requirements defined (Houmb, Georg, Jurjens, & France, 2007; Jurjens & Doser, 2005). It is also capable of verifying whether the system designed prevents previously designed attacks that will compromise the information system. This proposal has recently been extended to enable it to use patterns that support the modelling and verification of formal aspects of security. This proposal has recently been adapted to UML 2.0 (Jurjens & Schmidt, 2011).

With regard to the use of this approach in real situations, it is possible to highlight, for example, a methodology for the development of critical security systems using security use cases (Popp, Jurjens, Wimmel, & Breu, 2003). It is also possible to highlight the application of this model in the search engines used in the Intranet of a German automobile company, in which the security of these distributed security systems is analysed (Best, Jurjens, & Nuseibeh, 2007).

UMLSEC has a methodological approach focused on security requirements engineering.

### 3.1.2.6. Model Drive Security

The MDS proposal (Basin, Doser, & Lodderstedt, 2003) and its various applications (Alam, Breu, & Hafner, 2007; Oladimej, Supakkul, & Chung, 2007) is a clear example of the integration of software engineering and security engineering. The aforementioned studies present a model driven approach with which to obtain secure information systems. It is based on the generation of high level abstraction models which include security properties and the use of tools to automatically generate the system architecture. This is done by using model transformation techniques, including security properties and their corresponding security mechanisms. This is carried out by adding primitives and rules that make it possible to integrate the security into the development process in which the information system is designed. This
proposal employs SecureUML (Best et al., 2007; Lodderstedt, Basin, & Doser, 2002) for the purpose of system modelling. These models make it possible to integrate security aspects independently of the technology used and to detect and correct possible design errors. Several works based on the development of transformations that allow equivalence to be automatically obtained between SecureUML and UMLsec have appeared in recent years (Matulevicius & Dumas, 2011), and various evolutions of this proposal have also been carried out (Gunawan, Herrmann, & Kraemer, 2009; Gunawan, Kraemer, & Herrmann, 2011; Sánchez, Molina, García-Molina, & Toval, 2009; Schreiner & Lang, 2008).

It does not implement the cloud computing paradigm nor is it focused on the migration of legacy systems.

3.1.2.7. GSyA Proposals

The GSyA Research Group is working, among many others topics, on information systems security along with approaches related to secure methodologies or processes for the construction of various information systems. Some of these approaches are summarised below.

3.1.2.7.1. PWSSec

PWSSec - Process for Web Services Security (Gutiérrez, Fernández-Medina, & Piatini, 2005) has been created to facilitate and orientate the development of WS-based security systems. This process specifies how to define security requirements for WS-based systems, describes a logical security service-based reference security architecture and explains how to instantiate it to obtain a concrete security architecture based on the current WS security standards. This process can be used either in the development of a new system, since it can be perfectly integrated into the stages into which functional generic services-based and WS-specific (Endrei et al., 2004) development processes are divided, or it can be used for an already designed or developed system based on WS.

3.1.2.7.2. M-BPSec

The goal of the M-BPSec method (Method for Business Process Security) (Rodríguez, Fernández-Medina, & Piattini, 2007) is to perform the incorporation of security requirements into the business process specification. The process elicits security requirements which are captured through the application of an extension of the UML activity diagram, UML 2.0-AD (Rodríguez, Fernández-Medina, & Piattini, 2006b). The use of BPsec-Profile (Rodríguez, Fernández-Medina, & Piattini, 2006a), makes it possible to represent the business analyst’s perspective with regard to security. A new point of view with a high level of abstraction is therefore obtained, and this complements existing perspectives concerning security. M-BPSec is composed of four stages: Construction, Security requirement incorporation, Refining and Transformation.
3.1.2.7.3. **SREPPLine**

The Security Requirements Engineering Process for software Product Lines (SREPPLine) (Mellado, Fernandez-Medina, & Piattini, 2007; Mellado, Fernández-Medina, & Piattini, 2008) is a standard-based process that describes how to integrate security requirements into the software engineering process in a systematic and intuitive manner, along with a simple integration with the other requirements and the different phases/processes of the SPL development lifecycle. This process will also facilitate the fulfilment of the IEEE 830:1998 standard (IEEE, 1998), and it will help develop IS which conform to the aforementioned security standards with regard to the management of security requirements without it being necessary to know those standards perfectly, thus reducing the participation of security experts. In order to attain these goals, this approach is based on the reuse of security artefacts which are integrated into the variability model of the SPL, by providing a Security Core Assets Repository (SCAR), together with the integration of the Common Criteria (CC) (ISO/IEC 15408) into the SPL lifecycle. SREPPLine is composed of two sub-processes with their respective activities: PLSecDomReq (Product Line Security Domain Requirements Engineering sub-process) and PLSecAppReq (Product Line Security Application Requirements Engineering sub-process). These sub-processes cover the four basic phases of requirements engineering according to (Kotonya & Sommerville, 2000): requirements elicitation; requirements analysis and negotiation; requirements documentation and requirements validation and verification.

3.1.2.7.4. **MGSM–PYME**

This approach (Sánchez Crespo, 2009) has the aim of providing advances in ICT security management, particularly in the case of small and medium–sized enterprises. It is a methodology called the Methodology for Security Management and has matured in small and medium–sized enterprises (MGSM–PYME). There is also a realistic, pragmatic and agile model that makes it possible to evaluate and improve security management and its maturity level in information systems of small and medium–sized enterprises. This model is based on the most important international standards and regulations. The development of this methodology is based on a new approach of security management in small and medium–sized enterprises adapted not only to the size of the enterprise but also to its maturity level, using the ISO/IEC27002 standard (formerly ISO/IEC17799) as a reference framework. The basic idea of this methodology is that of the adaptation of the costs to the dimension of the enterprise. The methodology has been divided into three sub-processes which are as follows: Schemas generation (GEGS), ISMS generation (GSGS) and ISMS maintenance (MSGS).

3.1.2.7.5. **SecMobGrid**

This approach (García Rosado, 2009) defines a development process for secure mobile Grid systems (called SecMobGrid process - Secure Mobile Grid systems) that allows the integration of software engineering, to develop systems based on mobile Grid computing, with security engineering that incorporates the security aspects of this kind of systems from the first stages of development.
This process is composed of several activities of which it is possible to highlight the following: the analysis activity (D. G. Rosado, Fernández-Medina, López, & Piatiini, 2009) during which all system requirements (especially security requirements) are captured and specified basing on specific use cases, which are defined using a UML profile (GridUCSec-profile) (David G. Rosado, Fernandez-Medina, & Lopez, 2011; D. G. Rosado, Fernández-Medina, & López, 2009a, 2009b; D. G. Rosado, Fernández-Medina, López, & Piattini, 2010); the design activity, which is focused on the construction of a security architecture (D. G. Rosado, Fernández-Medina, & López, 2011) for this kind of systems from the reference security architecture developed (SMGridArch) that defines the minimum set of security services covering all possible security requirements that can be specified in mobile Grid systems; and the construction activity that is in charge of the implementation of the system using the existing tools for the development of Grid systems.

3.1.2.7.6. **Model-Driven Development of Secure OLAP Applications**

This approach (Bueno, 2009) presents the development of secure data warehouses focused on the OLAP technology, by means of including the security in an intermediate layer to be used by OLAP applications to access to the data warehouse information (Belén Vela, Blanco, Fernández-Medina, & Marcos, 2012; B Vela et al., 2013).

This proposal is integrated into a previously-defined architecture which adopts the model driven engineering paradigm in order to model and automatically generate the secure implementation of the data warehouse repository according to a relational approach. To achieve this purpose, the models needed to integrate the proposal into this previously-defined architecture, along with the suitable mechanisms required to automate the process to obtain the final and secure implementation for OLAP applications, are developed. Moreover, given that there are numerous legacy OLAP applications, this approach defines and integrates a modernisation process into the architecture which facilitates the improvement of existing OLAP applications, since it automatically obtains their corresponding models, improves them and automatically re-implements the improved system.

3.1.2.7.7. **Methodology for the Construction of Secure Information Systems Based on Patterns and Experience**

This proposal (Ortiz et al., 2010; Ortiz, Moral-Rubio, Garzás, & Fernández-Medina, 2011) consists of a methodology oriented towards the construction of information systems, covering all aspects of security from the first stages, so as to obtain a reliable, maintainable and robust product.

This approach presents a methodology with which to construct secure technological architectures for the information systems in this type of organisations. It is based on Security Patterns. In order to make an appropriate use of the solutions provided in the Security Patterns format, the author has chosen the Enterprise Security Patterns modality, which consists of a type of patterns that can be used to represent complete technological architectures of information systems at different levels of abstraction. A tool with which to support the various activities in the proposed methodology is also presented in order to simplify its adoption by any
organisation. The activities of which this methodology is composed are: Planning, Viability Study, Analysis, Design, Construction, installation/Test and Maintenance.

3.1.3. Proposals for Cloud Computing Security

Although some of the aforementioned security methodologies tackle cloud computing, we have also included a section containing those which are more focused on the cloud.

3.1.3.1. Framework for Information Security Governance in Cloud Computing Services

In this approach (Rebollo Martínez, 2014), the authors propose a security governance framework that considers the particularities of cloud deployments (ISGcloud). The ISGcloud framework compiles existing published guidance works on the field, and groups them homogeneously to provide a model that is capable of delivering a Security Governance process for cloud services. ISGcloud is led by standards, signifying that its proposed activities tend towards existing security and governance standards, resulting in an alignment with actual best practices.

The governance processes in the ISGcloud framework are, in part, aligned with the proposal in the ISO/IEC 38500 standard. According to this standard, the governance cycle is composed of three processes: a) Evaluate the current and future uses of the IT; b) Direct the preparation and implementation of policies in order to ensure that the IT fulfil the business objectives, and c) Monitor the fulfilment of the policies and the performance of the planned actions. The authors have now added a fourth process, denominated as Communicate, which emphasizes the importance of diffusing the security knowledge needed in a governance framework.

ISGcloud additionally contains security processes that can be executed throughout the lifecycle of the service. From this perspective, the ISGcloud framework contains additional security components that make it possible to integrate the particularities of computation into the cloud. In this respect, the definition of security processes that are developed in each of the phases of the cloud computing service lifecycle has been considered. This has been achieved by proposing a lifecycle based on the ISO/IEC 27036 standard which contains security aspects in the presentation of services by the providers. This lifecycle is composed of the following phases:

1. Planning and Definition of Strategy
2. Analysis of Cloud Security
3. Design of Cloud Security
4. Implementation or Migration of the Service
5. Secure Operation of Cloud
6. Termination of Service

The intention of this lifecycle is to allow ISGcloud to adapt to any type of cloud computing deployment. The service may have a slightly different lifecycle, depending on the specific details of the implementation, and the activities and tasks contained in each of the
lifecycle are, therefore, sufficiently flexible to adapt to the needs of any organisation. The four processes in the Security Governance form one of the dimensions of ISGcloud, and the six stages of the cloud computing service lifecycle become a second dimension. The framework is therefore designed from a bi-dimensional perspective, which represents that the iterative cycles of the governance processes (Evaluate-Direct-Monitor) are executed successively throughout the various stages of the cloud service, thus making it possible to guarantee that all the necessary aspects of security are appropriately fulfilled.

3.1.3.2. **Cloud Security and Privacy**

The authors of the book ‘Cloud Security and Privacy’ (Mather, Kumaraswamy, & Latif, 2009) list the aspects of security that they consider to be most relevant in the cloud computing environment. The perspective adopted is fairly revealing, thus facilitating the reader’s understanding and fomenting confidence that s/he is handling the most critical elements of security.

After introducing the most relevant aspects of cloud computing, together with the service models and means of implementation, the authors underline the following security-related questions in these environments:

- Infrastructure Security.
- Data and Storage Security.
- Identity Management and Access.
- Security Management.
- Privacy.
- Auditing and Conformity.

In order to be able to manage the aforementioned aspects, the authors propose developing a strategy in levels, in which the higher layers are focused on Security Leadership and Security Governance. Leadership encompasses the promotion and sponsorship of security initiatives in cloud computing environments, the identification of use cases and the definition of an appropriate strategy for the type of cloud service.

In an attempt to implement a security framework of these characteristics, the book proposes the use of the Open Security Architecture, which groups together the functions described (Open Security Architecture, 2008). This architecture defines a pattern with which to illustrate the principal functions of security in cloud computing, the principal roles that participate in vigilance and the minimisation of risks, collaboration through various organisations and those security controls that require particular emphasis.

3.1.3.3. **Cloud Computing Adoption Framework**

The Cloud Computing Adoption Framework (CCAF) (Chang, Kuo, & Ramachandran, 2016) is a comprehensive model for the systematic adoption and application of cloud security
principles. It is a conceptual framework, similar to ITIL version 3, with which to guide organisations towards the best practices. This framework can additionally be integrated with cloud computing services in order to provide the adopting organisations with added value. It is also an architecture framework focused on the delivery of a security service, in the form of developing a multi-layered security for data centres.

CCAF security aims to cover the following categories:

- Application software security, which deals with how to build systems that can automatically protect themselves.
- Network (LAN, MAN, GAN), wireless network security and platform security include Operating systems, virtualisation and systems software.
- Convergence network security, in which multi-network media infrastructures, social networks and technologies converge, this being one of the emerging areas of research.
- Service-oriented security, which concerns issues related to system services such as denial of service attacks, distributed denial of services, and web services.
- Cloud security, which deals with services security, data security and privacy in an attempt to ensure that services are delivered and assets are protected.
- Open-source software security, which includes issues such as trust, certification and qualification models.
- Software components and architecture security, which deals with building secure components and architectures that can be used as plug-ins.
- Web services security, which is essential as regards ensuring that secure services are delivered with integrity.
- Systems & Software security engineering, which deals with building security in CCAF right from the requirements stage.

CCAF has a methodological approach that is focused on cloud computing. Its utility has been validated in real cases. Various applications with which to support CCAF are also being developed.

3.1.3.4. Cloud Security Using Service Level Agreements

According to (Luna, Taha, Trapero, & Suri, 2015), specifying security parameters in Service-Level Agreements (secSLA) is useful as regards establishing common semantics with which to provide and manage security assurance from two perspectives: the security level being offered by a cloud service provider (CSP), and the security level requested by a cloud customer.

Cloud secSLAs allow a CSP to describe implemented security controls, associated metrics (Trapero, Luna, & Suri, 2016) and committed CSP values for those metrics. From a customer perspective, secSLAs allow a more transparent view of the CSP security levels, while also providing information with which to monitor the fulfilment of the customer’s security expectations. Unfortunately, the lack of relevant cloud secSLA standards is a barrier to its adoption (Luna, Suri, Iorga, & Karmel, 2015).
This work is part of the EC FP7SPECS (Secure Provisioning of Cloud Services based on SLA management) and H2020 ESCUDO-CLOUD projects.

The work developed for secSLA is very interesting. It attempts to tackle the subject of cloud security from a methodological point of view and by including references to standards. It specifically makes reference to both the ISO standards and those developed by the CSA. It also includes the study of security evaluation metrics.

3.1.3.5. Security reference architecture for cloud systems

Reference architectures (RAs) are useful tools as regards understanding and building complex systems, and many cloud providers and software product vendors have developed versions of them. RAs describe (no implementation details) the main features of their cloud systems at an abstract level. Security is a fundamental concern in clouds and several cloud vendors provide security reference architectures (SRAs) in order to describe the security features of their services.

An SRA is an abstract architecture describing a conceptual model of security for a cloud system and provides a means to specify security requirements for a wide range of concrete architectures.

Its creators (Fernandez & Monge, 2014) propose a method with which to build an SRA for clouds defined using UML models and patterns, as shown in Figure 3-2.

They first identify threats by analysing its use case activities. Identifying cloud threats is, however, not sufficient; a means to describe how an attack is carried out is also needed, along with which cloud units are compromised. They have developed misuse patterns that describe how an attack (misuse of information) is performed from the viewpoint of the attacker. They have started building a catalogue of cloud misuse patterns which can be used to verify whether security patterns have been placed in the architecture to stop misuses of information.
The methodology contains a good number of security patterns that are specifically for clouds (Fernandez, Yoshioka, & Washizaki, 2015). This methodology is based on standards and focused on security in Cloud computing. The innovative nature of this methodology signifies that tools with which to support it have yet to be developed.

3.1.3.6. Trust Model for Measuring Security Strength of Cloud Computing Service

Trust model measures the security strength and computes a trust value. A trust value comprises various parameters that are necessary dimensions along which security of cloud services can be measured. Trust model acts as a benchmark and ranking service to measure security in a cloud computing environment. Evaluating a cloud service security is a necessity for any organisation moving towards the cloud.

In this methodology (Shaikh & Sasikumar, 2015) the authors have identified a comprehensive list of security parameters that are necessary and sufficient to measure security with regard to the cloud computing environment. These parameters are incorporated into the trust model and the outcome is a trust value. This trust value can be a single value providing a notion of the overall security of a cloud service. It can also be broken down into various aspects of security based on the parameters and represented as a vector. A user can select a cloud service based on its requirement and demands concerning identity, data protection or any other measure listed in the trust value vector. Trust model consists of various parameters that depend on sub parameters and functions.

The parameters cover almost all aspects of security. The individual parameters are:

- Identity management-IDM: IDM is a key element of the security eco-system for cloud, and for any internet applications in general.
- Authentication: An authentication check is required to increase user confidence at the time of login and as an identity verification process.
- Authorisation: Users should not be able to carry out any actions for which they are not authorised. This property can be checked against the strength of authorisation.
- Data Protection: The crucial asset of both a user and any organisation moving on to the cloud is data. Data privacy issues are great concerns while moving data to and from a cloud environment.
- Confidentiality: A cloud service should protect the secrecy of the communication between a cloud user and provider and all other actions performed in various activities.
- Communication: Data or messages passed in the cloud computing environment are prone to eavesdropping or leakage. The communication strength measures the provision provided by the cloud service at the time of data or message transmission.
- Isolation: The multi-tenant nature of the cloud computing infrastructure leads to the problem of isolation of resources among multiple users. Security breaks and violations are the key factors that are mainly caused by isolation.
- Virtualisation: The concept of cloud computing is incomplete without the virtualisation feature. A virtualised infrastructure is more prone to attacks than a physical one. Techniques should be provided to secure the virtualised environment.

- Compliance: A compliance approval indicates that the method and process of a particular cloud service have been quantified by the known and authorised agencies. The security of a cloud computing service provider and a service can also be determined by the approval or certification from various compliances or standards.

  Trust based evaluation is proposed in the form of trust model. Trust value is the output of the trust model that measures the security strength.

  This proposal is highly focused on measuring the security in cloud computing. It is supported by tools and has been validated in real cases.

3.1.3.7. Other proposals

This section includes other cloud computing service security-related proposals that, in spite of not being as representative as those mentioned previously, contain concepts relevant to the content of this thesis.

3.1.3.7.1. Security and Control in the Cloud

The following authors (Julisch & Hall, 2010) use the indications concerning Information Security management in the ISO/IEC 27001 standard as a basis to propose the creation of a virtual system for security management, in which they extrapolate the concepts of the standard to the cloud computing environment. This proposal is directed at management organs, as regards both customers and providers, who may benefit from their recommendations on the risk management of externalised assets and new security measures and controls.

The authors point out that the security schemas normally adopted by cloud computing service providers are insufficient for their customers’ needs. In spite of the fact that various providers have security certificates, they are audited very infrequently, signifying that customers receive information that the system is working correctly at specific moments, but not during the intermediate period. It would be desirable to install mechanisms that would allow customers to carry out these checks in real time. Moreover, it is common to trust security to the clauses in the Service Level Agreement (SLA), but these clauses tend to be beneficial for the providers since sanctions for non-fulfilment are, generally, moderate.

3.1.3.7.2. Addressing cloud computing security issues

These authors’ proposal upholds that the use of a third trusted party would help to improve the security of cloud computing services and preserve the confidentiality, integrity and authenticity of the information (Zissis & Lekkas, 2012). This third party would be an entity that
would facilitate the safe interaction between customers and the provider, who all would trust. This third trusted party would, therefore, be in charge of providing the following functions:

- **Low and high-level confidentiality.** The third party would provide coded protocols (SSL, IPSec, etc) that would improve the confidentiality of the information.
- **Authentication of customer and server.** It is vital to verify the identity of users, servers and any other elements involved.
- **Creation of security domains.** The federation of domains makes it possible to establish trusting relationships among entities. This federation consists of a group of entities that share a set of security policies and access rules.
- **Cryptographic separation of data.** The authors propose combining the use of symmetric and asymmetric ciphering techniques in order to provide both stored data and that in transit with security.
- **Authorisation based on certificates.** Traditional identification models are not valid, and the authors therefore propose employing certificates with a greater number of attributes, such that decisions regarding the granting of permission will be able to take into account a larger set of variables.

3.1.3.7.3. **Cloud Security Issues**

One of the fundamental elements for the security of cloud computing services is the Service Level Agreement (SLA), which defines the provider’s provision of the service to the customer. The authors therefore propose the definition of a standardised procedure for its creation (Kandukuri, V, & Rakshit, 2009). SLAs normally contain information regarding the definition of the services, performance management, problem management, the sharing of responsibilities, existing guarantees, recovery from disasters and the conclusion of the service. The paper proposes that the following security elements, that will permit their standardisation and provide greater homogeneity, should also be added:

- **Privileged user access.**
- **Legal conformity.**
- **Data location.**
- **Recovery from disasters.**
- **Long-term availability.**
3.2. Proposals regarding the Migration of Legacy Systems to the Cloud

This section presents the methodologies and proposals found in literature for the migration of legacy systems to the cloud. We shall first revise the various proposals that exist for the migration of legacy systems. We shall then go on to analyse the proposals for migration to cloud computing. Finally, we shall analyse the various proposals with which to carry out a secure migration to cloud computing.

3.2.1. Methodologies for the Migration of Legacy Systems

This section presents the methodologies and proposals found in literature for the migration of legacy systems.

3.2.1.1. Renaissance

The aim of the Renaissance project (Warren, Fu, King, & Chang, 1999) was the development of a systematic approach for reengineering and software evolution. The focus was the efficiency and feasibility of methods in an industrial environment. In order to meet these requirements, the project partners developed an abstract process model for software evolution and software reengineering. This was tested on different systems. The model is oriented towards industrial development principles and models.

Furthermore, it supports technical evolution in addition to mechanisms for the migration of programming languages (3GL to 4GL) or platforms/architectures, e.g. the transition from mainframe to client-server architectures. A further object of the project was the identification and assessment of cost models so as to enable appropriate evolution strategies, process planning and risk assessments. The developed methodology is based on an incremental evolution. The integration of an existing legacy functionality is provided via distributed architectures.

The Renaissance project approach represents a pattern for the model-driven methodology, along with its implementation. Unfortunately, the Renaissance process model could not be reused here since it is highly generic.

This method is focused on migration. However, it is not focused on security issues, nor is it based on the cloud computing paradigm.
3.2.1.2. **MODELPLEX**

The target of MODELPLEX was similar to that of Renaissance: improving quality and productivity during the development and evolution of software systems. The central topic was the transfer of results into an industrial environment.

Part of the project was the development and application of a model-driven methodology (Bézivin, Barbero, & Jouault, 2007). This included the development of tools, the design of methods, the analysis, synthesis, verification, and validation of models, and the study of abstraction.

The original goal of the project, the development of software, had less in common with the migration of software systems. However, important outcomes of the project are the development of a knowledge model similar to KDM, the ATL transformation language and the Eclipse tool MoDisco. It provides a framework with which to create model-driven tools for the modernisation of existing software systems. The meta-model used for this approach is the KDM.

The modernisation of a system is described as:

- the extraction of information,
- the understanding of it and
- the transformation into new artefacts to support the modernisation.

MoDisco provides meta-models for the description of existing systems, discoverers for the automatic creation of models and further tools for programme understanding and transformation.

This is a very complete methodology for the migration of legacy systems. It is supported by one of the tools most frequently used for this type of tasks: MoDisco.

3.2.1.3. **SOAMIG**

The objective of SOAMIG - Migration of legacy software in service-oriented architectures (Fuhr et al., 2012) is to define an adaptable iterative migration process model. The SOAMIG process has four organisational phases that set out important milestones in migration projects. The phases collate several disciplines, highlighting activities during migration.

1. Preparation: The starting point of every migration project is legacy code which has to be prepared and standardised in the Pre-Renovation discipline by means of various reengineering activities in order to alleviate conversion activities.

2. Conceptualisation: 70%-75% of the activities in reengineering projects are independent of detailed project needs. A broad automation by eligible migration factories would appear to be possible. One of the central activities in migration projects is that of assessing the feasibility of migration and the applicability of the tool sets provided during Technical Feasibility.

3. Migration: The entire system is migrated after setting up a general migration strategy and tool support.
4. Transition: Code migration usually leads to hardly maintainable code, which requires additional reengineering.

Seven SOAMIG core disciplines are performed during the conceptualisation phase for a small part of the legacy system and eventually in the migration phase for the entire system, as shown in Figure 3-3:

This proposal has a methodological approach that is highly appropriate for the migration of legacy systems. However, it has not been adapted to the cloud computing paradigm, nor does it consider security as a fundamental element in migration.

3.2.1.4. Model-Driven Software Migration: A Methodology

The Methodology for Model-Driven Software Migration (Wagner, 2014) is a method for the model-driven migration of software systems.

The method consists of five phases: transformation of the source code into models, model analysis, abstraction by means of model transformation, splitting and the migration of the existing system and code generation.

The first part – the transformation of the programme code – includes the development and application of tools from the fields of compiler construction and programme analysis. The aim is to convert the source code into a machine readable form. The result is a representation of a control flowgraph (code-model), which is visualised graphically by means of a modelling tool. Model analysis (second phase) improves the understanding of the application. These analyses are based on the code-models and can be flexibly adapted to the specific project situation. This includes the creation of new ones, along with the integration of external tools.
The transition to the model level occurs in the third phase: a model abstraction step is applied, thus enabling information to be classified in the code-models and abstracted into a new model. The abstraction works on the programming interfaces of the underlying libraries and is therefore called API-based abstraction.

The first three stages form the basis of the subsequent migration of the system (step four). The migration is the remodelling of the existing software as a process model. The information obtained through the application understanding will guide this step. The process model developed is also partially associated with the functionality of the existing system. After completing this step, fully executable source code is generated from the migrated models (step five).

This proposal is supported by tools. It is highly appropriate for the migration of legacy systems, but is not, however, focused on the cloud computing paradigm, nor does it consider security as a basic element.

3.2.1.5. MARBLE: Modernisation Approach for Recovering Business Processes from Legacy Information Systems

MARBLE is a technique and a tool that supports business process archaeology by retrieving business processes from legacy source code (Ricardo Pérez-Castillo et al., 2011). MARBLE utilises an extensible, ADM-based framework to recover business processes.

To achieve this: (i) the information is collected in and is used from standard KDM (Knowledge Discovery Metamodel) (ISO/IEC, 2012) repositories and (ii) the information from KDM repositories is used to retrieve business process models (Ricardo Pérez-Castillo, Cruz-Lemus, de Guzmán, & Piattini, 2012).

MARBLE focuses on the reverse engineering stage of the re-engineering process. It proposes four abstraction levels (with four different kinds of models), in addition to three model transformations between them, in order to cover the whole path of the business process archaeology method between legacy information systems and business processes. The four generic abstraction levels proposed in MARBLE are the following:

- **Level L0.** As the lowest level of abstraction, L0 represents the legacy information system (LIS) in the real world as a collection of different software artefacts (e.g. source code, database, documentation).

- **Level L1.** This level consists of several specific models, i.e. one model for each different software artefact involved in the archaeology process (e.g., source code, database, user interfaces). These models are considered to be PSM (Platform-Specific Models) since they depict the software artefacts according to their specific technology or platforms.

- **Level L2.** This consists of a common PIM (Platform-Independent Model) which represents the integrated view of the set of PSM models at L1. The standard KDM metamodel is used for this purpose, since it makes it possible to model all the artefacts in the legacy system in an integrated and technologically independent manner.

- **Level L3.** As the highest level of abstraction, L3 represents a computational independent model of the system. It depicts the business processes retrieved from the knowledge concerning legacy information systems represented in the KDM repository at L2.
Business process models at L3 are represented according to the BPMN (Business Process Model and Notation) metamodel (O. OMG, 2011).

The four abstraction levels and the three model transformations among them are shown in Figure 3-4:

![Figure 3-4. MARBLE, a framework to support business process archaeology](image)

MARBLE provides a Java parser with which to obtain code models, which are transformed and integrated in a model repository according to the KDM standard.

The KDMs are then transformed into business process models by applying business pattern recognition. Finally, the tool allows the discovery, visualisation and editing of business process models. An in-depth description of the framework's functionality and capabilities is provided in (R Pérez-Castillo, de Guzmán, Ávila-García, & Piattini, 2009; Ricardo Pérez-Castillo et al., 2011).

The proposal described is of great interest to us for our proposal. It is based on standards and is supported with tools. Despite the fact that it is not based on the cloud computing paradigm and that security is not a fundamental element, it could be adapted owing to the use of the KDM standard.

3.2.1.6. IBUPROFEN: Improvement and BUusiness Processes Refactoring OF Embedded Noise

Business process models derived via the reverse engineering approach followed by MARBLETM often require some refinement before they can be utilised for further transformations. The IBUPROFEN (Improvement and BUiness Processes Refactoring OF Embedded Noise) approach has therefore been developed, which introduces a set of algorithms for the refactoring of business process models expressed in BPMN (Fernández-Ropero, Pérez-Castillo, & Piattini, 2013). It introduces a set of ten refactoring algorithms which can be applied to business process models represented by graphs, expressed in BPMN. These ten refactoring algorithms are divided into three categories regarding their purpose, namely: maximisation of relevant elements, fine-grained granularity reduction and completeness.
An overview of the refactoring performed by each of these algorithms is provided in (Fernández-Ropero, Pérez-Castillo, Cruz-Lemus, & Piattini, 2013). The process model refactoring algorithms introduced by IBUPROFEN are shown in Figure 3-5.

Figure 3-5. Process model refactoring algorithms introduced by IBUPROFEN

This proposal compliments the previous one. It is of great interest to us for our proposal since it provides a higher level of abstraction by introducing the BPMN models.

3.2.2. Methodologies for Migration to the Cloud

This section presents the methodologies and proposals found in the literature for migration to the cloud.

3.2.2.1. MODACLOUDS

The MODACLOUDS project (Ardagna et al., 2012) proposes a Model Driven approach whose objective is to support system developers and operators as regards exploiting multiple clouds for the same system and migrating part of their systems from cloud to cloud as needed.

The MODACloud platform relies on a domain specific language for the design and execution of applications on multiple clouds. The Model-Driven Engineering approach adopted by the MODACloud platform allows developers to build the system at various levels of abstraction. The three envisioned levels are: the Cloud-enabled Computation Independent Model (CCIM) to describe an application and its data, the Cloud-Provider Independent Model (CPIM) to describe cloud concerns related to the application in a cloud-agnostic manner, and the Cloud-Provider Specific Model (CPSM) to describe the cloud concerns needed to deploy and provision the application on a specific cloud.
Each layer of the architecture contains various models that can be manipulated within the MODACloud environment.

The models for the Cloud-enabled Computation Independent layer are: the Requirements Model that completes and formalises the service functional description; the Service Definition Model that describes the software to be developed as a set of components or services; the Data Model that describes the main data structures associated with the software to be; the Usage Model that specifies the way in which users are expected to exploit the functionality of the software to be, and the Service Orchestration Model that describes the behaviour of the glue between components and services.

The models for the Cloud-Provider Independent layer are: the Design Alternative and Deployment Model that describe the assignment of application components to underlying resources and the design alternatives and constraints that will drive the search for optimal solutions performed by design time exploration tools, and the Data Model that describes the data model in terms of logical models as a flat model, a hierarchical model and a relational model.

The models for the Cloud-Provider Specific layer are: the Design Alternative and Deployment Model that describes the assignment of application components to underlying resources and the design alternatives and constraints that will drive the search for optimal solutions performed by design time exploration tools that depend on the characteristics of the cloud resources of a specific cloud provider, and the Data Model that describes the data model based on the specific data structures implemented by the cloud providers.

This methodology is specifically for migration to Cloud computing. It would, however, benefit from a greater focus on security.

3.2.2.2. DEVAs

The approach presented in (Collazo-Mojica & Sadjadi, 2011) utilises the MDE methodology as a modelling approach that simplifies cloud architecture design, in addition to achieving platform independence from IaaS providers. By designing DEVAs (distributed ensembles of virtual appliances) non-expert users can easily architect interdependent virtual appliances on Infrastructure as a Service clouds. DEVA models include quality of service (QoS) constraints, which can account for the non-functional requirements of the modelled architecture. The approach proposes two DEVA metamodels with the main difference being that one allows the instantiation of resource-independent (RI-DEVA) models, and the other allows resource-dependent (RD-DEVA) models.

The distinction between resource dependent and independent models makes it possible to separate the concern of modelling a DEVA and the concern of modelling the resources needed to run that DEVA. The definition of a resource independent model allows a user to let DEVA framework allocate the required resources on the basis of the specified high-level policies and constraints. This can be achieved by transforming an RI-DEVA into a RD-DEVA using model to model (M2M) transformations.

The transformation between these models can be done in various ways. The task is to translate quantifiable constraints and policies to available resources in an IaaS cloud.
This methodology was specifically developed for migration to cloud computing. Although it includes several security elements, security is not considered as a fundamental element.

3.2.2.3. ARTIST

ARTIST (Bergmayr et al., 2013) proposes a software modernisation approach covering business and technical aspects. In particular, ARTIST employs Model-Driven Engineering techniques to automate the reverse engineering of legacy software and forward engineering of cloud-based software in such a way that modernised software truly benefits from targeted cloud environments.

ARTIST proposes a model-based software modernisation process that consists of a pre-migration, migration, and post-migration phase. Before the migration is performed, the Legacy Software is analysed in the pre-migration phase concerning the technical and non-technical consequences of possible migration strategies. This analysis results in well-defined Migration Goals constituting the input for the decision-making as regards how the migration should be performed in the subsequent phases. In a first step of the migration phase, models are reverse-engineered from the legacy software. These Legacy Platform-Specific Models comprise all the specifics imposed by the platform the legacy software is built on. in order to enable the coverage of a wide range of current and future modernisation scenarios and the reuse of reoccurring platform-independent migration patterns throughout various modernisation scenarios, the legacy PSM is transformed into a higher-level representation, denominated as a PIM (Platform-Independent Model). The PIM abstracts from platform-specifics, such as software runtime environments and data management capabilities. These platform-specifics need to be adapted to cloud providers’ offerings, as their cloud environments are typically unique and operate on different virtualisation layers, i.e. from infrastructure to platform to software as a service.

The PIMs are then subjected to model transformations, which are selected on the basis of the migration goals defined in the pre-migration phase. These transformations carry out the actual migration by applying optimisation patterns and integrating cloud-specific modernisation opportunities. As a result, model-based representations of the migrated software that comprise platform-specifics compatible with the selected cloud environment are produced. This “Cloudified” PSM is transformed into the executable migrated software hosted in a cloud environment.

In the post-migration phase, the model-based representations of legacy software and migrated software are employed to derive Equivalence Tests. Their objective is to verify that the migrated software behaves as expected. Furthermore, non-functional properties are evaluated to certify whether the migration goals are fulfilled.

This is achieved by analysing the execution of migrated software in order to obtain quality Measures, which are checked against the migration goals defined.

This is one of the most frequently referenced proposals for migration to cloud computing. It has a good methodological approach and permits the validation of the results obtained.
3.2.2.4. The CloudMIG Approach: Model-Based Migration of Software Systems to Cloud-Optimised Applications

The proposal by (Sören Frey & Wilhelm Hasselbring, 2011) presents a specific model with which to migrate legacy systems to the cloud, denominated as CloudMIG. The model consists of six activities whose objective is to transform the legacy system into a cloud-ready one; namely: extraction, selection, generation, adaptation, evaluation and transformation.

The work does not deal with security issues, although the 3rd activity (Generation) provides a model containing the target architecture violations of the cloud environment constraints (CEC). However, they are generic constraints, and the paper is not specific about the security constraints of either of the legacy or the target, and although it can be assumed that security constraints could be modelled as CEC, no indication is provided about how to do so.

It does not relate the model to any security standard or any process aimed at ensuring the security requirements of the target or the artefacts produced, although the Knowledge Discovery Metamodel (KDM) is proposed in the first activity as a suitable model with which to build the original legacy software architecture.

3.2.2.5. Cloudward Bound: Planning for Beneficial Migration of Enterprise Applications to the Cloud

The creators of Cloudward Bound (Hajjat et al., 2010) present a model that can be used to deploy enterprise applications in a hybrid cloud environment, in which applications are partly hosted on the organisation’s premises. It provides a detailed model to evaluate which components of a legacy application can be migrated to the cloud, and which components must stay. The authors illustrate the model with a real use case at a large university involving the migration of their Enterprise Resource Planning (ERP) system.

The approach specifically deals with the migration of access control lists and their definition in the firewalls of the organisation. It does not, however, deal with the security requirements of the application, but rather the security requirements of communications (firewall contexts and ACL rules).

The model is not software engineering but rather systems oriented, and neither security requirements nor migration from the application perspective are, therefore, addressed. The proposal does not mention any security standard.

3.2.2.6. REMICS-REuse and Migration of Legacy Applications to Interoperable Cloud Services

REMICS (Parastoo, Jørgen, Sadovykh, Barbier, & Benguria, 2010) presents a methodology to support the organisation in the adaptation of their legacy systems to the cloud. The methodology is characterised by a set of activities areas to be usually covered in this kind of initiatives and a recommended lifecycle that helps the organisation to manage with the lack of knowledge in this kind of projects.
The methodology covers seven activity areas:

- Requirements: focused in the additional requirements.
- Recover: focused in the recovery of the application logic from the legacy code.
- Migrate: focused in the definition and implementation of the migrated system, usually include the implementation of the SOA.
- Validate: focused in the implementation of validation activities over the migrated system.
- Supervision: focused in the implementation of monitoring and support features.
- Interoperability: focussed in the identification of interoperability issues and their solution.
- Withdrawal: focussed in the stop of the service in a managed way.

REMICS presents a tool-supported MDA methodology for use in migrating legacy applications to Cloud.

REMICS does not deal specifically with security, risks or any security standard. The authors refer readers to the REMICS project web page for further publications and the state of the art, but we have been unable to trace any further progress.

3.2.2.7. Other proposals

There are numerous proposals related to migration to the cloud, such as CMotion, CloudGenius, CloudStep or the cloud adoption toolkit.

The authors of “Decision Support Tools for Cloud Migration in the Enterprise” (Khajeh-Hosseini, Sommerville, Bogaerts, & Teregowda, 2011) propose a model for risk assessment when migrating legacy applications to the cloud and attempt to identify a template for risk evaluation and mitigation approaches. The paper does not propose a detailed legacy application migration process, nor does it deal with the security constraints of legacy apps or mention any security standard. Nevertheless, the authors show five risks associated with security according to a survey that they carried out of 50 academic papers.

The paper “Service Migration in a Cloud Architecture” (Kaisler & Money, 2011) presents several security and integration issues related to the migration of existing applications within three support areas: acquisition, implementation and security. The security section shows questions that the practitioner should ask him or herself, such as the fact that “data must often be retained locally to satisfy regulatory requirements” or regulatory and legal issues related to the security of data and/or its availability. It does not present a technical process for migration nor align any activity with the security process or with any security standard.

The creators of the “Dynamic Service and Data Migration in the Clouds” approach (Hao, Yen, & Thuraisingham, 2009) present a framework that can be used to ease the migration of services to the cloud and provide a decision model with which to select the services that can be moved. It does so from the point of view of the supporting SOA services, in which the general computing platform is also regarded as a service that uses VM technologies.
The proposal “Migrating Legacy Applications to the Service Cloud” (Zhang, Berre, Roman, & Huru, 2009) consists of a process with which to migrate legacy software into software-as-a-service architecture in seven steps. It relies on model driven architecture (MDA) transformations and the Software Engineering Institute horse model (Seacord, Plakosh, & Lewis, 2003). The seven steps are the following: architectural representation of the legacy, redesign of the architecture, MDA transformation, web service generation, web service-based invocation of legacy functionalities, selection of the CC platform and web service deployment. The model is applied to a case study concerning the migration of a legacy application in the area of oil risk analysis. The paper does not deal with security issues except in the last step (migration to the cloud) and security is mentioned only in a general non-specific manner, along with scalability and networking. Nor does it seem to ask any detailed questions about the security constraints of the legacy. The paper does not mention any security standard or any activity, task or process related to integration, or review the security constraints of the target.

3.2.3. Methodologies for Secure Migration to the Cloud

This section presents the methodologies and proposals found in literature for secure migration to the cloud.

3.2.3.1. CloudMIG extension

The objective of the CloudMIG approach is to support SaaS providers when semi-automatically migrating legacy software systems to the cloud.

The paper “An Extensible Architecture for Detecting Violations of a Cloud Environment’s Constraints During Legacy Software System Migration” (Soren Frey & Wilhelm Hasselbring, 2011) presents an extensible architecture for the detection of a legacy software system’s violations of constraints imposed by cloud environments when considering a migration. It is an important constituent of the CloudMIG approach which supports reengineers when migrating existing software systems to the cloud. Here, the detection and highlighting of crucial system parts is an essential early phase activity. Violations may be easy to fix, but a reenginer has to be aware of them. Furthermore, this approach supports the assessment of the severity of a detected violation. It is based on KDM and supports the use of certain tools such as MoDisco.

The process deals with risks related to the migration process, namely: the runtime constraints, but not specifically the security constraints of the legacy or the target platform. It does not relate the model to any security standard or any process aimed at ensuring the security requirements of the target or the artefacts produced.
3.2.3.2. Framework for cloud computing adoption: a roadmap for SMEs to cloud migration

A stepwise framework (Khan & Al-Yasiri, 2016) for cloud adoption has recently been formulated which identifies and provides recommendation for four of the most predominant challenges that are damaging the cloud industry and discouraging SMEs from the use of cloud computing, along with guiding SMEs towards successful cloud adoption. Moreover, this framework streamlines the cloud adoption process for SMEs by removing ambiguity as regards fundamentals associated with their organisation and the cloud adoption process.

The challenges identified above affect cloud adoption by SMEs and need to be addressed with adequate solutions and recommendations. All of the above problems are directly related to a bigger problem: the lack of a framework for SMEs that directs the process of migration to clouds. A framework for SMEs could eventually provide an answer to these problems in a structured manner so as to expedite SMEs’ adoption of the cloud.

This framework is divided into a three-stage migration process; the cloud preparation stage (CPS), the cloud requirement stage (CRS) and the cloud migration stage (CMS).

The cloud computing migration process requires a continuous improvement process prior to migration to ensure the efficiency of the system (or solution), and it is for this reason that the framework works in a circular loop to provide firm and positive results. An As-Is and To-Be business process is therefore adopted, in which the Cloud Requirement stage (CRS) of the framework is To-be, the Cloud Preparation Stage (CPS) is As-is and the Cloud Migration Stage (CMS) is the transformation. As the CPS continues, the goal should be set to prepare the enterprise to adopt cloud computing, and regular modification needs to be made to refine the process of getting the SME ready for cloud computing. These refinements will be vital in deciding the efficiency and sustainability of the system. As the process enters a CMS stage, it is ready to be transformed as per the proposed solution provided (or selected) for an enterprise.

The Cloud Preparation Stage (CPS) contains Assessing Organisation Readiness. The followings are four vital factors which cover the readiness of an organisation: Governance, Risk Assessment, Standards and Data classification and responsibilities.

This proposal is interesting because security is introduced in one of its stages. However, it could be improved if security were a fundamental part of the entire methodology and not just of one of its stages.

3.2.3.3. Impact of Critical Infrastructure Requirements on Service Migration Guidelines for the Cloud

A high level of information security in critical infrastructure IT systems and services has to be preserved when migrating their IT services to the cloud. Various legislative and security constraints often have to be met in line with best practice guidelines and international standards in order to perform the migration. To support the critical infrastructure providers in migrating their services to the cloud (Wagner, Hudic, Maksuti, Tauber, & Pallas, 2015) are developing process based migration guidelines for critical infrastructure providers, focusing on information security.
The methodology proposes a process-based guideline (model) for secure service migration towards the cloud environments. It includes a “Secure cloud migration taxonomy” and a “Secure cloud migration life cycle”.

If this taxonomy is to be used in an effective way manner, it must be incorporated into a process that pays attention to the information security aspects of cloudification. The authors propose a new approach for a Cloudification Security Development Life cycle (CloudSDLv1) of IT services which is based on a common security development life cycle. The approach for CloudSDLv1 comprises five phases, as shown in Figure 3-6:

The research has been funded by the H2020 project PRISMACLOUD and FP7 project SECCRIT. This proposal is very interesting since it is based on standards. It was published recently and is in an early development phase.

3.2.3.4. Secure migration to the cloud—in and out

According to (Kemmerich, Agrawal, & Momsen, 2015), the challenges for using a cloud service provider are security, network access, interoperability, and preserving IT-Know-How. They present an overview of the most important challenges and describe how to ensure that these challenges are dealt with.

They propose a detailed Five-Phase-Model for sustainable migration to the cloud:

- Planning: the planning phase is the most important phase because the prerequisites for all following phases and the accompanying procedures will be defined during the planning phase. Mistakes or uncertainties during this phase can affect the whole migration and operation.
3.3. Proposals regarding the Migration of Legacy Systems to the Cloud

- Contracts: the strategy and the cloud policy must be used as a basis to negotiate the contracts with both the CSP and the ISPs.
- Migration: the migration phase is the most complex phase. A previously conducted business and security analysis is used as a basis for the development of the implementation and Migration Security Concept (MSC).
- Operation: the operations phase is a more or less steady-state situation in which the cloud customer mostly has to take care that the quality of the IT-Service provision is sufficient. Measures described in the ISO 9000 (Quality Management) and ISO 27000 (Information Security Management) families have to be applied to guarantee the service quality required. Independent audits have to be carried out in order to guarantee the service quality defined.
- Termination: the termination phase is necessary since the rollback to internal IT-Service provisioning or the change of the CSP is not usually considered by a cloud customer in long-term planning.

The Five-Phase-Model is depicted in Figure 3-7:

![Five-Phase-Model](image)

Figure 3-7. Five-Phase-Model of secure migration to a Cloud Service Provider

This is a very interesting proposal since it is based on standards and a complete methodological approach is available. It could, however, be improved with the inclusion of a better study of legacy systems, to which no reference is made.

3.2.3.5. A security approach for Data Migration in Cloud Computing

Data migration to a cloud computing environment (Kushwah & Saxena, 2013) is, in many ways, an exercise in risk management. Both qualitative and quantitative factors apply in an
analysis. The risks must be carefully balanced against the available safeguards and expected benefits, with the understanding that accountability for security remains with the organisation. Too many controls can be inefficient and ineffective, if the benefits outweigh the costs and associated risks. An appropriate balance between the strength of controls and the relative risk associated with particular programmes and operations must be ensured.

Cloud Migration is one of the many points discussed as regards cloud managers confronting extreme problems at the time of data migration from a company’s server to a server that forms a cloud elsewhere. A discussion on why they confront these problems will be shown as follows. Moreover, if data migration does not take place systematically and properly, it can give rise to problems concerning data and the cloud security of a company’s assets that primarily consist of data.

This proposal presents a solution to the securing of the data migration process. The process consists of transitioning all or part of a company’s data, applications and services from on-site premises behind the firewall to the cloud, where the information can be provided over the Internet on an on-demand basis. While a cloud migration may have numerous challenges and raise security concerns, cloud computing can also enable a company to potentially reduce capital expenditures and operating costs while also benefiting from the dynamic scaling, high availability, multi-tenancy and effective resource allocation advantages that cloud-based computing offers.

This is an interesting proposal that is focused on the migration of data to cloud computing. The proposal could be broader to allow not only the migration of data but also that of the complete information system.

3.2.3.6. Other proposals

This section presents other proposals found in literature for secure migration to the cloud.

3.2.3.6.1. Secure Instance Migration Module (SIMM).

The SIMM (Syed, Musa, Rahman, & Jan, 2015) module is a Trusted Computing data protection mechanism for those instances during which data is migrating from one of the cloud provider’s zones to another. OpenStack has been customised to add the SIMM module hooks before performing the actual migration. This research is a work in progress and consists of securing other cloud services that include the movement of data. The SIMM module is the first step towards secure data migration in the cloud.
3.2.3.6.2. Security Migration Requirements

The paper ‘Security Migration Requirements: From Legacy System to Cloud and from Cloud to Cloud’ (Hussein, Hashem, & Li, 2013) discussed data migration from an organisation’s computers to the cloud, including the migration process between clouds along with workload migration between different kinds of clouds.

3.3. Comparison of Proposals

This section contains a comparative study of the proposals identified in the spheres of both information systems security and the migration of legacy systems to the cloud.

All of the proposals summarised in this study of the state of the art in this area provide information that is relevant from certain perspectives, and could thus be used as a basis for new security frameworks or as an extension of those that already exist.

As a part of the Systematic Literature Review described in Section 2.3, the data extracted from the relative studies identified were used as the basis to carry out a comparative analysis. This analysis considered the two spheres of knowledge in which the aforementioned proposals are classified with the objective of defining appropriate comparison criteria and reducing any bias in the process. These criteria have served to evaluate the strengths and weaknesses of each proposal in a systematic and easily understandable manner.

The proposals have been compared by employing the following criteria:

1. It is a complete methodological approach, from the definition of requirements to implementation.
2. The prominent consideration of security from the early stages.
3. Adaptation to the cloud computing paradigm.
4. It permits migration from legacy systems.
5. It is based on the standards most frequently accepted by industry.
6. It explicitly includes analysis and risk management.
7. Automated support.
8. It has been validated in real cases.
9. It permits the valuation and evaluation of security and associated risks.
10. It incorporates processes that take advantage of reuse.
11. It incorporates techniques regarding traceability among artefacts.
12. It is focused on agile development techniques.

The result of applying the comparison criteria defined to the proposals identified in this state of the art are summarised in the following table. This table indicates whether each proposal totally (“Y”), partially (“P”) or does not (“N”) fulfil each of the security criteria.

Table 3-1. Summary of proposals analysed
<table>
<thead>
<tr>
<th>PROPOSAL</th>
<th>Methodological approach</th>
<th>Considers security</th>
<th>Adapted to cloud</th>
<th>Migrate legacy systems</th>
<th>Based on standards</th>
<th>Analysis &amp; Risk Management</th>
<th>Automated support</th>
<th>Validated in real cases</th>
<th>Valuation &amp; evaluation</th>
<th>Reuse</th>
<th>Traceability</th>
<th>Agile techniques</th>
</tr>
</thead>
<tbody>
<tr>
<td>Secure Tropos</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>CLASP</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>SDL</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Touchpoints</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>UMLSEC</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Model Drive Security</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>ISGcloud</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>Cloud Security and Privacy</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>CCAF</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>Cloud secSLA</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>Security Reference Architecture</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>Trust model</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Renaissance</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>MODELPLEX</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>SOAMIG</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Model-Driven Software Migration</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>MARBLE</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>IBUPROFEN</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>MODACLOUDS</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>DEVAs</td>
<td>Y</td>
<td>P</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>ARTIST</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>CloudMIG</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Cloudward Bound</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>REMICS</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>CloudMIG extension</td>
<td>Y</td>
<td>P</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Framework for cloud computing adoption</td>
<td>Y</td>
<td>P</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>CloudSDL</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Secure migration to the cloud—in and out</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Data Migration</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
</tbody>
</table>

3 State of the Art
The existing security proposals for the development of information systems generally provide a methodological approach that is correct as regards dealing with security. However, they have not been adapted to the cloud.

The proposals regarding security in the cloud are focused on the cloud computing paradigm, and some of them are even based on standards. However, none of them are oriented towards the migration of legacy systems.

With regard to the migration of legacy systems, none of the proposals analysed have a cloud-migration approach nor do they tackle aspects of security – and particularly not aspects of security in the cloud.

The approximations studied for migration to the cloud do consider the cloud computing paradigm, but do not include aspects of security.

The proposals most similar to our objective are those concerning secure Migration to the cloud. However, none of them fulfils all the criteria demanded. Some of them are not aligned with standards. Others do not cover the entire process, for example they don’t include analysis and risk management. And those most similar to our process are in an incipient state and they are neither validated with real cases nor have automated support.

Although none of these proposals meets all the required criteria, some of them may be used together in a coordinated manner as part of a new proposal.

The SMiLe2Cloud proposal fulfils all of the criteria demanded. It has a complete methodological approach, from the definition of requirements to implementation. It considers security as a fundamental element in all stages, from the definition of the requirements of the legacy systems to migration to cloud computing. It is aligned with the principal standards most frequently accepted by industry. And finally, the methodology has been validated in a real case with the help of automated tools.
4. SMiLe2Cloud
This chapter contains the main contribution of this thesis: the SMiLe2Cloud (Secure Migration of a Legacy information system to the Cloud) methodology.

The chapter is structured as follows: We first provide an overview of the SMiLe2Cloud framework, indicating its most important features and its participants’ roles, after which the model is shown within the framework as regards its activities, tasks and artefacts; and finally, all the activities and tasks proposed in the framework are detailed.

4.1. SMiLe2Cloud overview

In this section, we propose a process (denominated as SMiLe2Cloud, as shown in Figure 4-1) for the secure migration of Legacy Information Systems (LIS) to the cloud model which is, on the one hand, based on the Software Engineering Institute’s (SEI) horseshoe model (Seacord et al., 2003) and, on the other, on the Deming cycle of continuous improvement. The process follows the Software & Systems Process Engineering Metamodel Specification (SPEM) notation as closely as possible.

Figure 4-1. SMiLe2Cloud (smiletocloud.com)

Figure 4-2 shows the SMiLe2Cloud process. The process consists of five activities addressed by 16 security domains described in (Cloud Security Alliance, 2011).

The extraction activity is focused on the use of reverse engineering to extract security issues from LIS to a security model (SMiLe2Cloud model) defined for the migration process.

The second activity is the analysis of the security requirements, which is based on the extension of the Secure Tropos methodology (Mouratidis & Giorgini, 2007) for the cloud.

The design activity is focused on selecting the service model and the deployment model, and making the selection of the cloud provider on the basis of the CSA STAR (Security, Trust & Assurance Registry) (Cloud Security Alliance, 2014).

The deployment activity is focused on developing the deployment specification and implementing the system.

The fifth activity concerns carrying out the evaluation when the validation of the security model migrated is, and captures the new security issues to be incorporated into a new cycle of the process and into an analysis of the improvements and changes proposed for the cloud system.
4.1.1. Principal Characteristics

The principal and most prominent characteristics of the SMiLe2Cloud framework are shown as follows:

- **Iterative processes.** SMiLe2Cloud is executed in an iterative manner at both a task level and a global level, i.e. each task can be repeatedly developed until the proposed objective has been achieved, and the global cycle can, in turn, give place to repeated iterations that cover the life of the cloud computing service. A successive refinement of the output products of each process is carried out with each iteration.

- **Reusability of the process.** Given the practical approach used to design SMiLe2Cloud, the processes defined can be reused in other domains or contexts. This quality has been achieved by modelling the components using the SPEM 2.0 specification (OMG, 2008), which provides a meta-model for process engineering with the concepts required to model, publish, manage, interchange and carry out software processes. This reusability is also favoured by the interfaces defined in the model and the modularity of the process proposed.
• Reusability of the products. The products resulting from all the SMiLeCloud processes are stored in a common repository, which facilitates their reuse in the following iterative cycles. These products are, in turn, refined in successive iterations, leading to a continuously improved process. This repository can, in turn, serve as successful experiences for other projects with similar characteristics.

• Alignment with security standards and best governance practices. The SMiLe2Cloud framework complies with the most relevant security standards and best practices in cloud computing, employing tools that are well known by professionals. The principal security references are the following (more details will be provided when the description of their integration into the proposal shown in this thesis is presented):
  • ISO/IEC 27001 (ISO/IEC 27001, 2013), for those aspects related to management and security controls.
  • Cloud Security Alliance Security Guidelines. (Cloud Security Alliance, 2011), for those aspects specifically related to security in the cloud.

• Traceability and monitoring of the development of SMiLe2Cloud. SMiLe2Cloud provides a set of indicators and metrics that can be used to define a control panel, which facilitates information regarding the monitoring of the development of the process. This continuous monitoring by the organisation’s management facilitates the achievement of objectives.

• Flexibility as regards adaptation to any cloud computing service. The totality of the activities and tasks proposed in the SMiLe2Cloud framework are parametrizable and modifiable in order to enable their adaptation to the customer organisation’s real situation or to the cloud computing scenario considered.

4.1.2. Participating Roles

This section shows the principal roles that participate in the SMiLe2Cloud framework activities and tasks. Since the scope of the processes covers the entire organisation, we have chosen the most representative roles that must be directly involved in each process.

The translation of the roles defined to physical people should be carried out for each particular case according to the organisation’s situation or scenario. The flexibility of the proposed framework makes it possible for the functions of each role to be shared by different people in very large organisations, whilst it also allows the same person to take on various roles, as might occur in small companies.

The roles identified for SMiLe2Cloud are those shown in Figure 4-3. A description of each of the participating roles is shown as follows:

• Software engineer: the software engineer will be responsible for guiding the entire migration process. S/he has an in-depth knowledge of the complete process and is in charge of coordinating the various participating roles. The success of the methodology depends, to a great extent, on his/her relationship with the business manager.

• Business manager: formed of the organisation’s management personnel. These people are responsible for defining the operation, activities, tasks, etc. It is vital for this role to be aligned with the organisation’s management personnel if the migration of security is to be a success.
- Reverse engineering expert: who specialises in obtaining the requirements from the legacy software.
- Requirements engineer: who is in charge of creating the system requirements.
- Security requirements engineer: who is in charge of creating the system security requirements.
- Security expert: the person(s) in charge of the security of the system.
- Developer: the person(s) in charge of managing the software elements that support the computing services used by the organisation.
- Systems expert: the person(s) in charge of managing the infrastructures, communications, data storage, etc.
- Cloud expert: the person(s) in charge of managing the cloud infrastructure. Although this role could be integrated with that of the systems expert, it is important to highlight that this figure will grow in importance in organisations in the next few years.
- Auditor: the person(s) who audit the organisation’s processes and who are generally independent personnel who carry out other cloud computing service functions. They can be internal or external to the organisation.
- Cloud service provider: this role reflects the participation in the security processes of the provider who provides the cloud computing service, since their implication is required along with that of the customer.

Figure 4-3. SMile2Cloud roles represented according to SPEM 2.0.
4.1.3. **Global Schema**

A summarised view of the schematisation of the activities and tasks contemplated in the SMiLe2Cloud framework is shown in Table 4-1.

<table>
<thead>
<tr>
<th>Table 4-1. Global schema of the SMiLe2Cloud activities and tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1. Extraction Activity</strong></td>
</tr>
<tr>
<td>1.1 Define model on BPMN</td>
</tr>
<tr>
<td>1.2 Refactor model on BPMN</td>
</tr>
<tr>
<td>1.3 Define system requirements</td>
</tr>
<tr>
<td><strong>2. Analysis Activity</strong></td>
</tr>
<tr>
<td>2.1 Analysis of Security Requirements</td>
</tr>
<tr>
<td>2.2 Align Security Mechanism</td>
</tr>
<tr>
<td><strong>3. Design Activity</strong></td>
</tr>
<tr>
<td>3.1 Identification of Deployment and Service Model</td>
</tr>
<tr>
<td>3.2 Selection of cloud provider</td>
</tr>
<tr>
<td><strong>4. Deployment Activity</strong></td>
</tr>
<tr>
<td>4.1 Deployment Specification</td>
</tr>
<tr>
<td>4.2 Implementation</td>
</tr>
<tr>
<td><strong>5. Evaluation Activity</strong></td>
</tr>
<tr>
<td>5.1 Evaluation</td>
</tr>
<tr>
<td>5.2 Definition of new requirements</td>
</tr>
</tbody>
</table>

4.2. **Modelling the SMiLe2Cloud Framework**

The SMiLe2Cloud governance framework has been formally modelled by following the SPEM 2.0 (*Software & Systems Process Engineering Meta-Model*) specification developed by the OMG (*Object Management Group*) (OMG, 2008). This specification is used to define various system processes and their components, providing the concepts required for their management and installation. The use of this type of modelling makes it possible to provide a homogeneous and standardised representation of the processes by means of automated electronic repositories, and additionally facilitates the reuse of content using tools that are external to the SMiLe2Cloud framework. It is, therefore, possible to provide the framework with additional support by means of a widely accepted documentation format that supports its diffusion and use by different types of organisations.

The SPEM 2.0 specification adopts an object oriented perspective and is, therefore, very similar to the UML (*Unified Modelling Language*) notation (OMG, 2012), whose latest version has been adopted by the ISO as the ISO/IEC 19505 standard.
4.2.1. Modelling the Activities

In order to provide the modelling of the SMiLe2Cloud activities and tasks with a higher degree of detail, we have employed the SPEM 2.0 process specification. This specification was followed to carry out the modelling according to the notation proposed in Figure 4-4.

Activity (kind = Iteration): Name of the iterative Activity
TaskUse: Name of the Task
  ProcessPerformer {kind: primary}
  RoleUse: Role Name {kind: in}
  WorkDefinitionParameter {kind: in}
  WorkProductUse: Name
  WorkDefinitionParameter {kind: out}
  WorkProductUse: Name {state: state}
Steps
  Step: Name of the step
Guidance
  Guidance {kind: type}: Name

Figure 4-4. Notation used to model the activities and tasks according to SPEM

In accordance with this notation, the formal specification of each task includes the roles of the personnel who participate in it (RoleUse), the artefacts involved in both the input and the output of the process (WorkDefinitionParameter), the steps proposed for the execution of the task (Step), and the guidelines, techniques or practical improvements suggested in order to support its development (Guidance).

4.2.2. Reference Guidance

This section presents some guidance which will be used in the SMiLe2Cloud process.

4.2.2.1. Cloud security controls

The CSA Cloud Control Matrix (Cloud Security Alliance, 2016a) provides a control framework in 16 domains that are cross-mapped onto other industry-accepted security standards, regulations, and controls frameworks.

The Cloud Security Alliance (CSA) is one such organisation that aims to provide security assurance within cloud computing through the provision of various guidelines and standards. One of these is the CCM, which is a security control framework for cloud providers and customers. The CCM aims to guide cloud providers and customers in the assessment of security risks in cloud offerings. It provides a detailed analysis of security principles and concepts based on the “Security Guidance for Critical Area of Focus in Cloud Computing v3.0” (Cloud Security Alliance, 2011). The control framework consists of 16 domains, which are cross-referenced to industry-accepted security standards and regulations.
The latest version of the CCM is 3.0.1, which consists of 16 domains and 133 controls. Each control domain contains a specification describing the conditions and policies of the control. The architectural relevance of the control is provided by means of several fields that cover the cloud infrastructure; physical, network, compute, storage, application and data. The domain is applied to cloud service delivery models is using the standard SaaS, PaaS, IaaS (SPI) model, which informs the user which service models are affected in the case of a given control. Also, the applicability of the scope is provided by means of a comprehensive list of standards and regulations.

This guidance will be used in the Activity 3 to select the most appropriate cloud security controls for each security mechanism identified in the SMiLe2Cloud Analysis activity.

4.2.2.1.2. Reference Model of Information Assurance & Security (RMIAS)

The Reference Model of Information Assurance & Security (RMIAS) (Cherdantseva & Hilton, 2013) endeavours to address the recent trends in the evolution of the IAS.

The RMIAS has four dimensions:

- Security Development Life Cycle Dimension: illustrates the progression of IAS along the Information System Development Life Cycle (ISDLC)
- Information Taxonomy Dimension: outlines the characteristics of information being protected
- Security Goals Dimension: outlines the set of eight security goals, also referred to as the IAS-octave.
- Security Countermeasures Dimension: categorises security countermeasures.

SMiLe2Cloud Framework uses two of these dimensions: the security goals dimension and the security countermeasures dimension.

The security countermeasures dimension categorises four types of countermeasures:

- Technical countermeasures: this term refers to the technical means designed to achieve security goals. For example, identification, authentication and authorisation help to achieve integrity, confidentiality and accountability. Cryptography is one of the main security technologies that protect both integrity and confidentiality. Other examples of technical countermeasures are biometrics, digital signature, firewall, intrusion detection system, antivirus, etc.
- Organisational countermeasures: this refers to administrative activities which aim to build and maintain a secure environment in which the security countermeasures selected can be effectively implemented and managed. Examples of organisational measures are security strategy, security policy, procedures, governance, audit, compliance, business continuity and contingency planning, physical security, best practices, etc.
- Human-oriented countermeasures: these address the impact of the human-factor on IAS. Some authors argue that people play the most essential role in achieving security since the effectiveness of any technical or organisational security solution may be hindered if not supported by the individuals involved.
Legal countermeasures: this refers to the use of legislation for the purposes of information protection. Information often escapes the safe boundaries of an organisation, whether intentionally or otherwise. In such cases, neither technical nor organisational measures can help to protect information. Legal countermeasures play an important role in these situations. Some examples of legal measures are: established information ownership, legally agreed and enforced information classification and labelling schemes, service-level agreements, job contracts and employee non-disclosure agreements, law (e.g. copyright law), etc.

The RMIAS incorporates the IAS-octave as the Security Goals Dimension. The IAS-octave is a set of eight broadly applicable security goals:

- Accountability - An ability of a system to hold users responsible for their actions (e.g. misuse of information)
- Auditability - An ability of a system to conduct persistent, non-bypassable monitoring of all actions performed by humans or machines within the system
- Authenticity/Trustworthiness - An ability of a system to verify identity and establish trust in a third party and in information it provides
- Availability - A system should ensure that all system’s components are available and operational when they are required by authorised users
- Confidentiality - A system should ensure that only authorised users access information
- Integrity - A system should ensure completeness, accuracy and absence of unauthorised modifications in all its components
- Non-repudiation - An ability of a system to prove (with legal validity) occurrence/non-occurrence of an event or participation/non-participation of a party in an event
- Privacy - A system should obey privacy legislation and it should enable individuals to control, where feasible, their personal information (user-involvement)

The set of security goals is outlined in Table 4-2, which also shows the applicability of goals to the components of an IS.

<table>
<thead>
<tr>
<th>Security Goals</th>
<th>Information</th>
<th>People</th>
<th>Processes</th>
<th>Hardware</th>
<th>Software</th>
<th>Networks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accountability</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Auditability</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Authenticity/Trustworthiness</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Availability</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Confidentiality</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Integrity</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Non-repudiation</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Privacy</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
This guidance will be used in the Activity 3 to align the most appropriate cloud security controls for each security mechanism identified in the SMiLe2Cloud Analysis activity.

The cloud controls are aligned with the security mechanisms by means of security goals and security countermeasures. There are some controls related to each security goal and security countermeasure.

The list of security controls of CCM v3 aligned with RMIAS security goal dimensions and security countermeasure dimensions is outlined in Appendix II - List of security controls of CCM v3 aligned with RMIAS.

4.2.3. Modelling the Artefacts

This section presents the metamodel denominated as “SMiLe2Cloud model”, which guides the process of migrating to the cloud following the tasks defined in the SMiLe2Cloud framework.

4.2.3.1. SMiLe2Cloud model

This section presents the SMiLe2Cloud model. It commences with an explanation of the model. The model is a combination of existing concepts, based mainly on Secure Tropos (Mouratidis, 2004), and new concepts included specifically for the SMiLe2Cloud framework. Finally, the entire SMiLe2Cloud model is shown.

4.2.3.1.1. SMiLe2Cloud model overview

The SMiLe2Cloud model gives support to the SMiLe2Cloud framework in the process of migrating security features of a legacy system to a new environment as cloud computing.

The SMiLe2Cloud security model is a simplification of the model defined in Mouratidis et al. (Mouratidis, Argyropoulos, & Shei, 2016), since some of the concepts that are not essential for the SMiLe2Cloud framework, such as the soft goal, have been deleted, while certain other classes that address particular cloud security issues or address particular aspects of SMiLe2Cloud framework, such as cloud controls or security metrics, have been included.

This SMiLe2Cloud model is integrated in all stages of the framework SMiLe2Cloud.

4.2.3.1.2. SmiLe2cloud existing concepts

Secure Tropos (Mouratidis, 2004) combines concepts from requirements engineering in order to represent general concepts and security engineering so as to represent security-oriented concepts. The main concepts are shown in the Figure 4-5:
A goal represents a condition in the world that an actor would like to achieve (Yu, 1995). In other words, goals represent actors’ strategic interests.

An actor represents an entity that has intentionality and strategic goals within the multiagent system or within its organisational setting (Yu, 1995). An actor can be a human, a system, or an organisation.

In the context of cloud computing, it is also defined a special class for an actor; a cloud actor. A cloud actor is an actor that has two unique characteristics, it provides a deployment model and it supports a service model. It is worth stating that as an actor, a cloud actor also inherits all the attributes and associations of the actor. For example, it has goals and capabilities and it requires resources (Mouratidis, Islam, Kalloniatis, & Gritzalis, 2013).

There is also another special class of an actor: a malicious actor. A malicious actor’s intention is to introduce threats that exploit vulnerabilities into the system (Mouratidis et al., 2013).

A plan represents, at an abstract level, a way of doing something (Bresciani, Perini, Giorgini, Giunchiglia, & Mylopoulos, 2004). The fulfilment of a task can be a means to satisfy a goal.

A resource is a physical or informational entity required by one of the actors (Bresciani et al., 2004). The main concern when dealing with resources is whether the resource is available and who is responsible for its delivery.

The main concept introduced by Secure Tropos is a security constraint. The security constraints in the Secure Tropos methodology are used to represent security requirements (Mouratidis, 2011). A security constraint is a specialisation of the concept of constraint.

Security objectives represent a set of principles or rules that contribute towards the achievement of the system’s security (Mouratidis, 2011). These principles identify possible solutions to the security problems and they can usually be found in the form of the organisation’s security policy. Examples of these objectives are authorisation, integrity and availability.

A vulnerability is defined as a weakness or flaw, in terms of security and privacy, that exists in a resource, an actor and/or a goal (Mouratidis et al., 2013). Vulnerabilities are exploited by threats, as an attack or incident within a specific context. A Threat represents circumstances that have the potential to cause loss, or is a problem that can put the security features of the
system in danger (Mouratidis et al., 2013). Threats can be operationalised through the use of different attack methods, each of which exploits a number of system vulnerabilities.

An attack method in Secure Tropos is an action that aims to cause a potential security violation in the system (Mouratidis, 2011).

Security mechanisms represent standard security methods with which to help satisfy the security objectives (Mouratidis, 2011).

4.2.3.1.3. SMiLe2Cloud new concepts

The SMiLe2Cloud model includes the following new concepts, as shown in the Figure 4-6: cloud controls, security goals, security countermeasures, cloud migration metrics, security services and SLAs.

![Figure 4-6. SMiLe2Cloud model concepts](image)


The SMiLe2Cloud “security goals” and “security countermeasures” are defined in the Reference Model of Information Assurance & Security RMIAS. Secure Tropos identifies the security requirements (modelled as security constraints) and identifies a set of security objectives and a set of security mechanisms for each of them. Each security mechanism is related to one or more security goals defined in IAS-octave (Confidentiality, Integrity, Availability, Accountability, Auditability, Authenticity/Trustworthiness, Non-repudiation and Privacy) and one or more security countermeasures in the fourth dimension of the RMIAS (Organisational, Human-oriented, Technical and Legal).

Some controls are related to each security goal and countermeasure. The cloud controls are aligned with the security mechanisms by means of security goals and countermeasures. In other words, the cloud controls are implementations of the security mechanisms for the cloud.

The SMiLe2Cloud “metrics” are used to validate the security of the system. Once the entire process has been moved to the cloud in a secure manner, it is time to validate the security of the system. This activity is based on a set of cloud migration metrics. These metrics will
evaluate the availability of a security mechanism, the correctness of a security entity and the effectiveness level of a security mechanism.

The SMiLe2Cloud “Security Services” are the list of security services that it is necessary that the cloud actor implement in order to fulfil all the requirements previously defined.

The SMiLe2Cloud “SLAs” are the service-level agreements that should be defined to guarantee all the requirements previously defined will be achieved.

4.2.3.1.4. SMiLe2Cloud model artefacts

The SMiLe2Cloud model guides the SMiLe2Cloud process, it is integrated in all stages of the framework SMiLe2Cloud. The process starts with an empty model, and each activity add some concepts to the model. For each activity the SMiLe2Cloud model will be used as input and output, the main difference is that the output contains the new concepts introduced during the activity.

To clarify the different artefact that are input and output for each activity the artefact SMiLe2Cloud model has been divided into sub artefacts as shown in the Figure 4-7.

The sub artefacts identified are:

- System Requirements goal model: this artefact will be the output of the extraction activity and the input of the analysis activity. It contains actors, goals, plans, measures, mechanisms, resources, capabilities and constraints.
- System Security Requirements: this artefact will be used during the analysis activity. It contains the security mechanisms, security constraints, security goals, security objectives, vulnerabilities, attack methods, malicious actors and threats.
4.2 Modelling the SMiLe2Cloud Framework

- System Security Requirements aligned with CSA Domains: this artefact will be the output of the analysis activity. It contains security goals, security countermeasures and cloud controls.
- Cloud: this artefact will be used during the design activity. It contains the service model, the deployment model and the cloud actor.
- Metrics: this artefact will be used during all the process. In the analysis activity the metrics will be included and in the evaluation activity those metrics will be validated.
- Services: this artefact will be used in the deployment activity. It contains the security services and the SLAs.

Each activity of the process is defined in SPEM. In order to improve understanding of the model each activity will contain as input and output the artefact SMiLe2Cloud model together with the sub artefact that contains the concepts included in the activity. For example, the output of the extraction activity will contain the artefact SMiLe2Cloud model along with the sub artefact Security Requirements goal model (that is part of the SMiLe2Cloud model).

The SMiLe2Cloud model is shown in the Figure 4-8 (the background color of each concept represent the sub artefact to which it belongs).
Figure 4-8. The SMiLe2Cloud model
4.3. SMiLe2Cloud Activities and Tasks

This section contains a description of the activities and tasks that form the main structure of the SMiLe2Cloud framework. For each task, we provide information regarding the steps proposed in order to achieve satisfactory objectives during its execution.

4.3.1. Activity 1 Extraction

The extraction is the activity in which the system requirements for the LIS are derived from the code of the LIS. It is a reverse engineering process. The process is assisted by reverse engineering tools in order to ease the tasks and steps that the software engineers must perform to put the different requirements and controls in place.

As shown in Figure 4-9, the input is the Legacy Information System (LIS) and the output is the SMiLe2Cloud Model that contains the System Requirements goal model. The main stakeholders that are involved in this activity are developers, software engineers, business managers, reverse engineering experts and requirements engineers.

The proposed approach consists of a series of phases (see Figure 4-9), as listed below:

1. Extraction of BPMN process models from the source code of the legacy system using the MARBLE framework.
2. Refactoring of the extracted process model via the IBUPROFEN algorithms.

3. Process-to-goal transformation to create an initial goal model from the refactored BPMN process model.

We initially extract the BPMN process models from the source code of the legacy system in the "Define Model on BPMN" task. This is done using the MARBLE framework. The next task is the "Refactor Model on BPMN" task, during which we refactor the model extracted in the previous task via the IBUPROFEN algorithms. Finally, in the "Define system Requirements" task, we create an initial goal model from the refactored BPMN process model using a process-to-goal transformation algorithm.

This activity produces internal artefacts which represent outputs for some tasks and inputs for others. Figure 4-10 shows a graphical representation of the extraction activity tasks using SPEM 2.0 diagrams.

A detailed description of this activity, which we have considered in our process using the SPEM 2.0 textual notation, is provided below. We define its tasks, roles, steps, work products and guidance, which will be characterised according to the discipline to which they belong. We shall now briefly define each task in this activity, indicating the steps to follow in order to successfully execute these tasks.

4.3.1.1. Task 1.1 Define model on BPMN

MARBLE is a technique and a tool that supports business process archaeology by retrieving business processes from legacy source code (Ricardo Pérez-Castillo et al., 2011). MARBLE utilises an extensible ADM-based framework to recover business processes.

In order to achieve this: (i) the information is collected in and is used from standard KDM (Knowledge Discovery Metamodel) (ISO/IEC, 2012) repositories, and (ii) the information
from the KDM repositories is used to retrieve business process models (Ricardo Pérez-Castillo et al., 2012).

MARBLE focuses on the reverse engineering stage of the re-engineering process. It proposes four abstraction levels (with four different kinds of models), in addition to three model transformations between them, in order to cover the whole path of the business process archaeology method between legacy information systems and business processes.

The main objective of MARBLE is to provide a first version of business process models that, when compared with a business process redesigned by business experts from scratch, represents a more efficient solution and a good starting point from which to achieve a business process archaeology.

Figure 4-10 shows the Inputs, Outputs and steps of Task 1.1 according to SPEM 2.0. The input is the Legacy Information System (LIS), while the output is the BPMN model. There are three steps: Transformation L0 – L1, Transformation L1 – L2 and Transformation L2 – L3. This task is performed using the MARBLE reverse engineering tool.

![Figure 4-11. SPEM 2.0 view of Inputs, Outputs and steps of Task 1.1](image)

The three generic transformations between the four abstraction levels (L0, L1, L2 and L3) proposed in MARBLE are the following:

- Transformation L0-to-L1. This transformation obtains PSM models from each legacy software artefact. Classical reverse engineering techniques, such as static analysis, dynamic analysis, program slicing and dicing, formal concept analysis, subsystem decomposition, and so on, could be used to extract the knowledge from any software artefact and build the PSM model related to it. These PSM models are represented according to specific metamodels. For example, a Java metamodel may be used to model the legacy source code, or an SQL metamodel to represent the database schema, etc.
- Transformation L1-to-L2. The transformation between levels L1 and L2 consists of a set of model transformations carried out to obtain a PIM model based on the KDM
metamodel. This PIM model is built from the PSM models from level L1. The L1-to-L2 transformation can be implemented by means of QVT. The transformation from the legacy information system (L0) to the KDM model (L2) is not direct owing to the fact that, in many cases, the platform-specific knowledge in the intermediate level L1 might be used to infer more business knowledge. The semantic gap between the legacy system and its KDM model is, therefore, incrementally reduced through L1.

- Transformation L2-to-L3. This transformation is based on a set of patterns. When a specific structure is detected in the KDM model from level L2, each pattern indicates what elements should be built and how they are interrelated in the business process model in L3. This is known as pattern matching and can be implemented in MARBLE by means of QVT relations, the declarative part of the QVT language. In addition, this last transformation can be assisted by business experts who know the organisation. The external information provided by experts also serves as valuable knowledge in business process archaeology. Experts can determine inconsistent or incoherent fragments in the preliminary business process models obtained after pattern matching; they can refactor the business process models, and incrementally fit the process models to the real behaviour of the organisation.

The main stakeholders of this task are developers, reverse engineering experts and software engineers.

The following outputs will be obtained as a result of this task: a first version of the BPMN model that represents a more efficient solution and a good starting point from which to achieve a business process archaeology.

With regard to the techniques and practices required for the realisation of this task, we can use the KDM and BPMN to carry out reverse engineering and model the business processes. Those involved in the project can also attend meetings.

The formal specification of this task with SPEM 2.0 notation is shown in Figure 4-12.

| TaskUse: 1.1. Define model on BPMN |
| ProcessPerformer {kind: primary} |
| RoleUse: Developers {kind: in} |
| RoleUse: Reverse engineering expert {kind: in} |
| RoleUse: Software engineer {kind: in} |
| WorkDefinitionParameter {kind: in} |
| WorkProductUse: Legacy Information System (LIS) |
| WorkProductUse: Reverse Engineering Tools |
| WorkDefinitionParameter {kind: out} |
| WorkProductUse: BPMN Model {state: initial draft} |

Steps
- Step: Transformation L0 – L1: LIS to PSM.
- Step: Transformation L1 – L2: PSM to KDM.
- Step: Transformation L2 – L3: KDM to BPMN.

Guidance
- Guidance {kind: Practice}: Meetings
- Guidance {kind: Guideline}: OMG Knowledge Discovery Metamodel (KDM)
- Guidance {kind: Guideline}: OMG Business Process Model and Notation (BPMN)

Figure 4-12. Detailed description of the task 1.1 using SPEM 2.0.
4.3.1.2. **Task 1.2 Refactor model on BPMN**

Business process models derived via the reverse engineering approach followed by MARBLE often require some refinement before they can be utilised for further transformations. The IBUPROFEN (Improvement and BUSiness Processes Refactoring OF Embedded Noise) approach has, therefore, been developed for this purpose. This approach introduces a set of algorithms for the refactoring of business process models expressed in BPMN (Fernández-Ropero, Pérez-Castillo, & Piattini, 2013).

It introduces a set of ten refactoring algorithms, as shown in Figure 3 5 of the section 3.2.1.6, which can be applied to business process models represented by graphs, expressed in BPMN. These ten refactoring algorithms are divided into three categories regarding their purpose, namely: maximisation of relevant elements, fine-grained granularity reduction and completeness. An overview of the refactoring performed by each of these algorithms is provided in (Fernández-Ropero, Pérez-Castillo, Cruz-Lemus, et al., 2013).


Figure 4-13 shows the Inputs, Outputs and steps of Task 1.2 according to SPEM 2.0.

The main stakeholders of this task are developers, reverse engineering experts and software engineers.

As a result of this task, we will obtain the refactored version of the BPMN model obtained in the previous task.

With regard to the techniques and practices required for the realisation of this task, the BPMN can be used to model the business processes.

Figure 4-14 shows a detailed description of Task 1.2 using SPEM 2.0.
4.3.1.3. **Task 1.3 Define systems requirements**

A series of transformation rules needs to be defined in order to facilitate the transition from business process models, expressed in BPMN and derived from legacy source code using the MARBLE framework, to Secure Tropos goal models. This process-to-goal transformation will create an additional, higher level of abstraction, represented by a goal model of the legacy system. At this level of abstraction, it is easier for non-technical stakeholders to elaborate on the overall system security by defining certain easily comprehensible constraints. Such constraints can be captured by the Secure Tropos goal model and mapped back onto the process model in order to be implemented during the redesign of the legacy systems.

The proposed transformation will essentially derive a goal model, in which security will be elaborated and expressed using Secure Tropos.

Transformation rules have been defined, using concept mappings between Secure Tropos and BPMN concepts and providing instructions on how they can be utilised. These mappings are based on conceptual similarities between the paired concepts, identified after a semantic analysis of the formal documentation and meta-models of the two modelling approaches (Mouratidis & Giorgini, 2007; O. OMG, 2011). A transformation algorithm has been defined, which provides instructions on how these mappings can be utilised, in order to transform the refactored process model by IBUPROFEN to a Secure Tropos goal model. This algorithm is shown in the following table.

The table 4-3 shows the algorithm employed for the transformation process.
4.3 SMiLe2Cloud Activities and Tasks

Table 4-3. Algorithm for the transformation process

| Step 1 | For each lane \((l)\) of the process model:  
Create a corresponding actor \(a(l)\) in the goal model. |
|--------|---------------------------------------------------------------------|
| Step 2 | For each sub-process \((p)\) of the process model:  
Create a corresponding goal \(g(p)\) at the goal model.  
For each of the sub-activities \((p')\) of \(p\):  
Create a corresponding sub-goal \(g(p')\), within \(g(p)\). |
| Step 3 | For each data object \((d)\) of the process model:  
Create a corresponding resource \(r(d)\) at the goal model. |
| Step 4 | For each message exchange \((m)\) of the process model, between two activities \((p_s, p_r)\) in two different lanes \((l_s, l_r)\):  
Create a dependency link \(dl(m)\) at the goal model, from the dependent goal \(g(p_s)\) to the dependee actor \(a(l_r)\). |
| Step 5 | For each exclusive or inclusive gateway \((x)\) between sub-activities \((p_1, ..., p_n)\) of the process model:  
Create an OR or AND decomposition \(or(x)\) of the corresponding goals \((g(p_1), ..., g(p_n))\) at the goal model. |

The application of the above transformation rules in a process model derived using the MARBLE framework and refactored using the IBUPROFEN algorithms enables a basic Secure Tropos goal model to be produced. This basic goal model is the main input upon which the organisation’s stakeholders will carry out the security elaboration of the system.

Figure 4-15 shows the Inputs, Outputs and steps of Task 1.3 according to SPEM 2.0. The input of this task is the BPMN model obtained previously. The output of this task is the SMiLe2Cloud Model that contains the System Requirements goal model. There are five steps, which are defined in Table 4-3.

Figure 4-15. SPEM 2.0 view of Inputs, Outputs and steps of the task 1.3
The main stakeholders of this task are business managers, requirements engineers and software engineers.

As a result of this task, we will obtain a first version of the SMile2Cloud Model that will contain the specification of the system requirements goal model.

With regard to the techniques and practices required for the realisation of this task, we shall use the algorithm for the transformation process. We shall also use the SMile2Cloud Model to model the system requirements.

Figure 4-16 shows a detailed description of Task 1.3 using SPEM 2.0.

**TaskUse: 1.3 Define system requirements**

- **ProcessPerformer** (kind: primary)
  - RoleUse: Business manager (kind: in)
  - RoleUse: Requirements engineer (kind: in)
  - RoleUse: Software engineer (kind: in)

- **WorkDefinitionParameter** (kind: in)
  - WorkProductUse: BPMN Model
  - WorkProductUse: Reverse Engineering Tools

- **WorkDefinitionParameter** (kind: out)
  - WorkProductUse: Specification of System Requirements goal model
  - WorkProductUse: SMile2Cloud Model (state: initial draft)

**Steps**
- Step: Transformation process step 1
- Step: Transformation process step 2
- Step: Transformation process step 3
- Step: Transformation process step 4
- Step: Transformation process step 5

**Guidance**
- Guidance (kind: Template): SMile2Cloud Model Template
- Guidance (kind: Guideline): Algorithm for the transformation process

Figure 4-16. Detailed description of the task 1.3 using SPEM 2.0.

### 4.3.2. Activity 2 Analysis

This activity concerns the definition of the security requirements. The Secure Tropos methodology (Mouratidis & Giorgini, 2007) has, therefore, been extended for the cloud. Some cloud-specific concepts are introduced, such as cloud controls, security goals and security countermeasures.

As shown in Figure 4-17, the input is the SMile2Cloud Model that contains the System Requirements goal model, while the output is the SMile2Cloud Model that contains the System Security Requirements aligned with CSA Domains. The main stakeholders that are involved in this activity are requirements engineers, software engineers, business managers, security requirements engineers, security experts and auditors.
This activity is composed of two tasks: “Analysis of Security Requirements” and “Align Security Mechanism”. In the “Analysis of Security Requirements” task, the SMiLe2Cloud model is used to derive a set of security requirements with which the system must comply in the new environment. Security requirements are mainly obtained by analysing an organisation’s attitude towards security and after studying its security policy. The security requirements are modelled as security constraints and a set of security objectives and a set of security mechanisms is identified for each one of them. The objective of the “Align Security Mechanism” task is to map the security mechanism identified in the previous step onto the cloud controls specified in the Cloud Control Matrix.

The Figure 4-18 shows a graphical representation of the extraction activity tasks using SPEM 2.0 diagrams.
A detailed description of this activity, which we have considered in our process using the SPEM 2.0 textual notation, is provided below. We define its tasks, roles, steps, work products and guidance, which will be characterised according to the discipline to which they belong. We shall now briefly define each task in this activity, indicating the steps to follow in order to successfully execute these tasks.

### 4.3.2.1. Task 2.1 Analysis of Security Requirements

The SMiLe2Cloud model is used to derive a set of security requirements with which the system must comply in the new environment. Some requirements of the original LIS may no longer be applicable to the target system, since the cloud ecosystem might simply have made them redundant or unnecessary. It is also necessary to bear in mind that not all the cloud controls may be applicable to the LIS; an analysis of the applicability of new cloud requirements is, therefore, necessary before we can proceed further.

The analysis activity is a crucial part of the SMiLe2Cloud process as the security requirements are extracted and defined on the basis of the system requirements elicited from the previous activity. The input for this activity is the system requirements elicited during the previous extraction activity, which is based on the BPMN model and converted into the SMiLe2Cloud model as an Extensible Markup Language (XML) file. At this stage, the system requirements have already been elicited and have been modelled in the Secure Tropos tool. In order to model cloud security requirements, we extend the Secure Tropos methodology in order to incorporate cloud properties.

The system requirements are modelled using existing elements such as goal, actor, plan, resource and constraints in Secure Tropos, in order to visually represent the relationships and requirements of the legacy system. We then extend the Secure Tropos notation to capture the security requirements for cloud systems.

This task has four steps: identify security constraints, identify threats, identify security mechanisms and introduce them into the goal model. This task will be performed using the SecTro tool. Figure 4-19 shows the Inputs, Outputs and steps of Task 2.1 according to SPEM 2.0.
The main stakeholders of this task are business managers, requirements engineers, security requirements engineers, security experts and software engineers.

As a result of this task, we will obtain a new version of the SMiLe2Cloud Model that will contain the specification of system security requirements.

With regard to the techniques and practices required for the realisation of this task, Secure Tropos can be used to model the security requirements that will be part of the SMiLe2Cloud Model, and meetings and interviews with the people involved in the project can also take place.

Figure 4-20 shows a detailed description of Task 2.1 using SPEM 2.0.

TaskUse: **2.1 Analysis of Security Requirements**

- **ProcessPerformer** {kind: primary}
  - **RoleUse**: **Business manager** {kind: in}
  - **RoleUse**: **Requirements engineer** {kind: in}
  - **RoleUse**: **Security Requirements engineer** {kind: in}
  - **RoleUse**: **Security Expert** {kind: in}
  - **RoleUse**: **Software engineer** {kind: in}

- **WorkDefinitionParameter** {kind: in}
  - **WorkProductUse**: Specification of System Requirements goal model
  - **WorkProductUse**: SMiLe2Cloud Model {state: initial draft}

- **WorkDefinitionParameter** {kind: out}
  - **WorkProductUse**: Specification of System Security Requirements
  - **WorkProductUse**: SMiLe2Cloud Model

**Steps**
- **Step**: Identify security constraints
- **Step**: Identify threats
- **Step**: Identify security mechanisms
- **Step**: Introduce security objects in SMiLeModel

**Guidance**
- **Guidance** {kind: Practice}: Meetings
- **Guidance** {kind: Practice}: Interviews
- **Guidance** {kind: Guidelines}: Secure Tropos
- **Guidance** {kind: Template}: SMiLeModel Template
4.3.2.2. Task 2.2 Align Security Mechanism

The objective of this task is to map the security mechanism identified in the previous task onto the cloud security controls specified in the Cloud Control Matrix (see Section 4.2.3.1.1). This activity is performed by identifying the security goals and the security countermeasures defined in “The Reference Model of Information Assurance & Security” (RMIAS) (see section 4.2.3.1.2) for each security mechanism. The cloud controls are aligned with the security mechanisms by means of security goals and security countermeasures. There are some controls related to each security goal and security countermeasure.

Once the cloud security controls have been identified and the security mechanisms and the security controls included in SMiLe2Cloud model, we can add these mechanisms to the metrics repository for them to be evaluated in the evaluation activity.

The list of security controls of CCM v3 aligned with RMIAS security goal dimensions and security countermeasure dimensions is outlined in Table 4-4. This table outlines only a subset, and the entire table is shown in Appendix II.

<table>
<thead>
<tr>
<th>Security goal dimensions</th>
<th>Security countermeasure dimensions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authentication/Trustworthiness</td>
<td>Organisational/Legal</td>
</tr>
<tr>
<td>Accountability</td>
<td>Organisational/Legal</td>
</tr>
<tr>
<td>Confidentiality/Integrity/Availability/Accountability</td>
<td>Organisational/Technical</td>
</tr>
<tr>
<td>Auditability</td>
<td>Organisational</td>
</tr>
<tr>
<td>Accountability</td>
<td>Organisational/Legal</td>
</tr>
</tbody>
</table>

Table 4-4. Security controls aligned with RMIAS
Figure 4-21 shows the Inputs, Outputs and steps of Task 2.2 according to SPEM 2.0. The input of this task is the SMiLe2Cloud Model that contains the System Security Requirements obtained previously. The output of this task is the SMiLe2Cloud Model that contains the System Security Requirements aligned with CSA domains. There are three steps: identify security goals, identify security countermeasures and align security mechanism with CSA domains. This task will be performed using the transformation for the alignment of security mechanisms and cloud controls defined in Section 5.3.3.2.

The main stakeholders of this task are requirements engineers, security requirements engineers, security experts, cloud experts and software engineers.

As a result of this task, we will obtain a new version of the SMiLe2Cloud Model that will contain the specification of system security requirements aligned with the Cloud Control Matrix.

With regard to the techniques and practices required for the realisation of this task, the table defined in Annex II will be used to align security mechanisms with the security controls in the Cloud Control Matrix. The result of this alignment will be part of the SMiLe2Cloud Model. Meetings and interviews with the people involved in the project will also take place.

Figure 4-22 shows a detailed description of Task 2.2 using SPEM 2.0.
TaskUse: 2.2 Align Security Mechanism

ProcessPerformer {kind: primary}
  RoleUse: Requirements engineer {kind: in}
  RoleUse: Security Requirements engineer {kind: in}
  RoleUse: Security Expert {kind: in}
  RoleUse: Cloud Expert {kind: in}
  RoleUse: Software engineer {kind: in}

WorkDefinitionParameter {kind: in}
  WorkProductUse: Specification of System Security Requirements
  WorkProductUse: SMiLe2Cloud Model

WorkDefinitionParameter {kind: out}
  WorkProductUse: Specification of System Security Requirements aligned with CSA
  WorkProductUse: SMiLe2Cloud Model

Steps
  Step: Identify dimensions of security mechanism
  Step: Identify goals of security mechanism
  Step: Align security mechanism with CSA domains

Guidance
  Guidance {kind: Practice}: Meetings
  Guidance {kind: Practice}: Interviews
  Guidance {kind: Guideline}: CSA Cloud Control Matrix
  Guidance {kind: Guideline}: Table of security controls aligned with RMIA S
  Guidance {kind: Template}: SMiLeModel Template

Figure 4-22. Detailed description of Task 2.2 using SPEM 2.0.

4.3.3. Activity 3 Design

The design activity is focused on the selection of the service model and the implementation model, and using the STAR certificate (Cloud Security Alliance, 2014) in order to make a decision as to which cloud provider to choose.

As will be observed in the following figure, the input is the specification of security requirements obtained in the previous activity. The requirements specification is aligned with the CCM v3, signifying that it has a list of the controls with which our clouds provider should comply in order to achieve the level of security required.

The output is the SMiLe2Cloud Model including the cloud model (which contains the deployment model, the service model and the cloud provider).

The main stakeholders that are involved in this activity are cloud experts, software engineers, business managers and cloud service providers.
4.3 SMiLe2Cloud Activities and Tasks

The design activity consists of two tasks: “Identification of Implementation model and service model” and “Selection of Cloud provider”. The “Identification of implementation model and service model” task is focused on identifying the implementation and service models to which we wish to migrate, depending on the customer’s need, the fulfilment of the requirements, the resources available, etc. Once the implementation and service models have been selected, and the security requirements of the system aligned with the CSA domains are available, it is possible to select the cloud provider that, according to the STAR standard, best fits with the security needs. The STAR standard provides a list of controls that each cloud provider implements from those defined in the CCM v3 matrix.

The Figure 4-24 shows the tasks and steps in the SMiLe2Cloud process design activity using the SPEM notation.

A detailed description of this activity that we have considered in our process using the SPEM 2.0 textual notation is shown as follows. We define its tasks, roles, steps, work products
and guidance, which will be characterised according to the discipline to which they belong. We shall now briefly define each task in this activity, indicating the steps to follow in order to successfully execute these tasks.

4.3.3.1. **Task 3.1 Identification of Deployment and Service Model**

This task is focused on identifying the implementation and service models to which we wish to migrate, depending on the customer’s needs, the fulfilment of the requirements, the resources available, etc. The distinctions made by the NIST (National Institute of Standards and Technology) (NIST, 2009) are followed for this purpose. There are three service models in the Cloud (SaaS, PaaS and IaaS) and four implementation models for the Cloud (public, private, community and hybrid).

The result of this task is the identification of the most appropriate implementation model and service model as regards the characteristics of our legacy system, along with the resources available and the level of security that we wish to achieve in the new cloud system to which we have migrated. It is, therefore, aligned with the CCM so as to obtain the information required for the implementation model and the service model in order to best cover our security needs.

The only input for this task is the output artefact from the previous activity (see Figure 4-25), and this is the specification of the security requirements aligned with the CSA domains. The outputs generated by this task are the implementation and service models selected, which will be the input for the following task. The steps in this task are: identify implementation model and identify the service model using the security requirements specification of the system as a basis. The CCM can be used as a template to help in the selection of providers and controls, while the SMiLe2Cloud tool is used as support. Figure 4-25 shows the Inputs, Outputs and steps of Task 3.1 according to SPEM 2.0.

![Figure 4-25. SPEM 2.0 view of Inputs, Outputs and steps of the task 3.1](image-url)
The main stakeholders that are involved in this activity are software engineers, cloud specialists and business managers.

As a result of this task, we will obtain a new version of the SMILe2Cloud Model that will contain the service and the deployment model.

With regard to the techniques and practices required for the realisation of this task, we can use the NIST definition of the deployment and service models, and meetings and interviews with the people involved in the project can also take place.

Figure 4-26 shows a detailed description of Task 3.1 using SPEM 2.0.

4.3.3.2. Task 3.2 Selection of cloud provider

Once the implementation model and the service model have been selected and the security requirements of the system aligned with the CSA domains are available, it is possible to select the cloud provider that best fits with the security needs obtained from the STAR standard. The STAR standard provides a list of controls that each cloud provider implements from those defined in the CCM v2 matrix.

A list of the controls needed to cover the security requirements defined in the analysis activity is, therefore, available, as is the list of cloud providers who comply with those controls, and it is thus possible to identify those cloud providers who comply with the security requirements.

Once the list of cloud providers that comply with the security requirements has been obtained, one or another is selected according to other variables, such as cost, the technologies to be implemented, etc. This selection is carried out using a SWOT analysis.
The SWOT analysis is a management tool that facilitates the strategy planning process, providing the information required to implement corrective actions and measures, and for the development of improvement projects. The term ‘SWOT’ corresponds to the names of the four elements analysed during the development of the analysis: weaknesses, threats, strengths and opportunities.

In order to develop the SWOT matrix, it is necessary to select the strengths, opportunities, threats and weaknesses that may have the greatest impact on the organisation. These elements are characterised by considering economic, technical, etc., factors. It is recommended that it be developed by creating a workshop of experts and carrying out brainstorming activities.

The output of this task will be the most appropriate cloud provider that best fits with security requirements and controls that must be implemented in the destination system, together with a list of the necessary security controls that cover the security requirements analysed and specified in the previous activity (the analysis activity). The SMiLe2Cloud tool guides the design process.

The input artefacts are the two generated in the previous task – the implementation model and the service model. There will be two output artefacts: the most appropriate cloud provider selected that complies with the security requirements of the system, and the SMiLe2Cloud model, which contains the list of possible security controls that the providers should supply. The guidelines and templates used to support this activity will be the STAR register and the SMiLe2Cloud tool.

This task has three steps: identify the list of cloud providers that comply with the security requirement, carry out the SWOT analysis and select the cloud provider. This task will be performed using the transformation described in 5.3.4.2. Figure 4-27 shows the Inputs, Outputs and steps of Task 3.2 according to SPEM 2.0.

![Figure 4-27. SPEM 2.0 view of Inputs, Outputs and steps of the task 3.2](image)

The main stakeholders that are involved in this activity are the software engineer, the business manager, the cloud expert and the cloud provider.
As a result of this task, we will obtain the cloud provider that fulfils all the security requirements for the service and deployment model selected.

With regard to the techniques and practices required for the realisation of this task, we shall use the CSA STAR standard to select the cloud provider that best fits with the security requirements defined.

Figure 4-28 shows a detailed description of Task 3.2 using SPEM 2.0.

**Task Use: 3.2 Selection of cloud provider**
- **Process Performer** {kind: primary}
  - **Role Use**: Business manager {kind: in}
  - **Role Use**: Cloud Expert {kind: in}
  - **Role Use**: Cloud Service Provider {kind: in}
  - **Role Use**: Software engineer {kind: in}
- **Work Definition Parameter** {kind: in}
  - **Work Product Use**: SMiLe2Cloud Model
  - **Work Product Use**: Cloud
- **Work Definition Parameter** {kind: out}
  - **Work Product Use**: Cloud
  - **Work Product Use**: SMiLe2Cloud Model

**Steps**
- Step: Identify the list of cloud providers that meet security requirements
- Step: Perform SWOT analysis
- Step: Select cloud provider

**Guidance**
- Guidance {kind: Practice}: Meetings
- Guidance {kind: Practice}: Interviews
- Guidance {kind: Guideline}: CSA STAR
- Guidance {kind: Template}: SMiLe2Cloud Model Template

Figure 4-28. Detailed description of the task 3.2 using SPEM 2.0.

### 4.3.4. Activity 4 Deployment

The deployment activity is focused on developing the deployment specification and the implementation of the system.

As is shown in the Figure 4-29, the input is the SMiLe2Cloud Model that contains the service model, the deployment model, the cloud provider and the cloud security controls. The output is the SMiLe2Cloud Model with the Services, that includes the Security Services and the Service Level Agreements (SLAs) signed with the cloud provider. The main stakeholders that are involved in this activity are software engineers, business managers, cloud experts, security experts, system experts and the cloud service provider.
This activity is composed of two tasks: “Deployment specification” and “Implementation”. Once we have defined the security controls that our application must meet and we have the cloud provider to which we are migrating, the STAR standard will guide us while carrying out the “Deployment specification” task. Finally, the “implementation” itself takes place. During the implementation task, it could be necessary to contract the services and to sign the Service Level Agreement (SLA) and develop custom security elements.

The Figure 4-30 shows a graphical representation of the extraction activity tasks using SPEM 2.0 diagrams.
A detailed description of this activity that we have considered in our process using the SPEM 2.0 textual notation is provided as follows. We define its tasks, roles, steps, work products and guidance, which will be characterised according to the discipline to which they belong. We shall now briefly define each task in this activity, indicating the steps to follow in order to successfully execute these tasks.

4.3.4.1. Task 4.1 Deployment Specification

Once we have defined the security controls that our application must meet and we have the cloud provider to which we are migrating, the STAR standard will guide us while carrying out the implementation.

As mentioned in Task 3.2, the CSA STAR (Cloud Security Alliance, 2014) standard provides, for each cloud provider, the list of controls it implements from those defined in the CSA’s Cloud Controls Matrix (CCM) (Cloud Security Alliance, 2016a). Most of the major cloud providers are certified according to the STAR standard.

CSA STAR consists of three levels of assurance:

- **Level one:** CSA STAR Self-Assessment is a complimentary offering that documents the security controls provided by various cloud computing offerings. Cloud providers submit a completed Consensus Assessments Initiative Questionnaire (CAIQ), or submit a report documenting compliance with Cloud Controls Matrix (CCM). This information then becomes publicly available, promoting industry transparency and providing customer visibility into specific provider security practices.

- **Level two:** comprises attestation and certification. In each case, the STAR uses the CSA CCM to align with and complement an existing standard or reporting criterion.

- **Level three:** CSA STAR Continuous Monitoring enables the automation of the cloud providers’ current security practices.

Thanks to the Consensus Assessments Initiative Questionnaire (CAIQ) it is possible to identify the security services of the cloud provider selected on the basis of the CCM security controls defined in previous activities.

This task has two steps: align CCM security controls with the cloud provider’s security services and identify the selected provider’s security services. This task will be performed using the transformation described in 5.3.5.1.

Figure 4-31 shows the Inputs, Outputs and steps of Task 4.1 according to SPEM 2.0.
The main stakeholders of this task are software engineers, business managers, cloud experts, security experts, system experts and the cloud service provider.

As a result of this task, we will obtain the deployment specification that will include the security services of the cloud provider selected.

With regard to the techniques and practices required for the realisation of this task, we will use the CSA CAIQ to define the deployment specification for the provider selected.

Figure 4-32 shows the detailed description of Task 4.1 using SPEM 2.0.

**TaskUse: 4.1 Deployment Specification**

- ProcessPerformer {kind: primary}
  - RoleUse: Business manager {kind: in}
  - RoleUse: Security Expert {kind: in}
  - RoleUse: Cloud Expert {kind: in}
  - RoleUse: Systems expert {kind: in}
  - RoleUse: Cloud Service Provider {kind: in}
  - RoleUse: Software engineer {kind: in}

- WorkDefinitionParameter {kind: in}
  - WorkProductUse: SMiLe2Cloud Model
  - WorkProductUse: Cloud

- WorkDefinitionParameter {kind: out}
  - WorkProductUse: Services
  - WorkProductUse: SMiLe2Cloud Model

**Steps**
- Step: Align CCM security controls with security services of the selected provider
- Step: Identify security services of the selected provider

**Guidance**
- Guidance {kind: Guideline}: CSA CAIQ
- Guidance {kind: Practice}: Meetings
- Guidance {kind: Practice}: Interviews
- Guidance {kind: Template}: SMiLe2Cloud Model Template

Figure 4-32. Detailed description of the task 4.1 using SPEM 2.0.
4.3.4.2. **Task 4.2 Implementation**

Finally, the implementation itself takes place. During the implementation task it could be necessary to contract the services and to sign the Service Level Agreement (SLA) and develop custom security elements.

This task has three steps: contract the security services and sign Service Level Agreements (SLA), develop custom security services and set all the security controls in working conditions. Figure 4-33 shows the Inputs, Outputs and steps of Task 4.2 according to SPEM 2.0.

![Figure 4-33. SPEM 2.0 view of Inputs, Outputs and steps of Task 4.2](image)

The main stakeholders of this task are software engineers, business managers, cloud experts, security experts, system experts and the cloud service provider.

As a result of this task, we will obtain the Service Level Agreement (SLA) signed to guarantee that the cloud provider will fulfil all the security services.

With regard to the techniques and practices required for the realisation of this task, we will use the CSA CAIQ to implement the security services in the cloud provider selected.

Figure 4-34 shows a detailed description of Task 4.2 using SPEM 2.0.
4.3.5. **Activity 5: Evaluation**

Once the entire process has been moved to the cloud in a secure manner, it is time to validate the security of the system.

As shown in Figure 4-35, the input is the SMiLe2Cloud Model and the Services. The output of this activity is the SMiLe2Cloud Model that contains the new requirements that will be the input for the following iteration of the process. The main stakeholders that are involved in this activity are requirements engineers, software engineers, business managers, cloud experts, security experts and auditors.
The “Evaluation” task will initially help reflect the quality of the security mechanisms incorporated using metrics. At the end, in the "Definition of new requirements" task, a continuous watch on the levels and metrics of security is advisable by adding or updating new goals or constraints for the future iterations in the process.

Figure 4-36 shows a graphical representation of the extraction activity tasks using SPEM 2.0 diagrams.
A detailed description of this activity that we have considered in our process using the SPEM 2.0 textual notation is provided below. We define its tasks, roles, steps, work products and guidance, which will be characterised according to the discipline to which they belong. We shall now briefly define each task in this activity, indicating the steps to follow in order to successfully execute these tasks.

4.3.5.1. Task 5.1 Evaluation

The last activity of Smile2Cloud process, the evaluation activity, requires the use of the metric taxonomy proposed by Moussa et al. (Ouedraogo et al., 2013), which will help reflect the quality of the verification process in the Security Assurance. An evaluation of the metrics is defined for each of the security mechanisms identified in the analysis activity. These security mechanisms are identified in our SMiLe2Cloud model, which is an input of this task, and this set of metrics can then be used to evaluate the level of fulfilment and security capability of each mechanism used in the migration process.

Moussa et al. propose adapting the Systems Security Engineering Capability Maturity Model to represent the quality levels achievable by a security verification process, and some of the ISO/IEC 15408 (ISO/IEC, 2009) or Common Criteria’s (CC) families as quality requirements pertinent to Security Assurance.

The increasing level of assurance effort is, according to CC, based upon:

1. Coverage of the verification. The effort is greater because a larger portion of the security mechanism’s functionalities is included in the verification.
2. Depth. Verification efforts deployed to a finer granularity of the security mechanism (i.e. at its atomic properties level) are more reliable.
3. Rigour. The more structured and formal the verification, the more stringent the result obtained.
4. Independence of verification. Verification performed by a third-party evaluator or a software tool provides more assurance than a self-assessment.

The levels used to depict the quality of the process used to verify the security posture of IT products are based on SSE-CMM, and are:

- QL1 or Performed informally: depicts a verification process that is neither rigorously undertaken nor planned and tracked. It may be performed by a human expert who relies on individual knowledge of the security mechanisms.
- QL2 or Structurally performed: refers to an audit that is carried out according to rules and principles depicted in an internal procedure for the verification.
- QL3 or Structured and independent verification relates to a verification that is performed according to a well-defined process using an approved standard or procedures developed by a third party.
- QL4 or Semi-complete verification concerns a verification that follows a well-defined process with a usage of software tools, which cover most of the relevant parts (functions and properties) of the security mechanism.
- QL5 or Complete verification refers to a verification in which all known relevant parts of the safeguard are appropriately investigated in both depth and breadth.
The authors assign a set of capability levels to each of the metrics defined, as is shown in Figure 4-37 and which are explained below:

**Coverage** is defined as the extent to which the set of functionalities of a security mechanism that are relevant to the security of an IT system is examined during the verification. The capability levels for this metric are: QAM_COV.1, which corresponds to a verification that is carried out for functionalities not formally estimated to be important for the security mechanism as regards to performing its correct functionality; QAM_COV.2, which is attained when some of the key functionalities of the security mechanism are evaluated in the process, and the highest level, QAM_COV.3, which is assigned in cases in which all the security functionalities of the security mechanism deployed are verified in the evaluation process.

**Depth** of the verification is defined as the portion of the security properties, which are relevant to the well functioning of the security mechanisms, that are covered by the verification process. The capability levels for this metric are: QAM_DPT.1, which reveals that there is uncertainty as to whether those properties involved in the verification constitute all of the security properties, a part of them or none at all; QAM_DPT.2, which indicates a specification of what would constitute the set of security properties for the security mechanism that exists but none of the security properties defined have been verified during the process; QAM_DPT.3, which reveals that the set of security properties for the security mechanism is known but only some of them are examined during the verification process, and QAM_DPT.4, which, in contrast signifies that all known security properties for the security mechanism are verified.

**Rigour of verification** denotes the maturity of the verification process, whether it follows a systematic process and how sophisticated the means of verification are. The capability levels for this metric are: QAM_RIG.1, which is assigned to the verification processes that are not guided by a specification as to what parts of the security mechanism should be vetted and how this should be carried out. However, when such guidance on how and what to verify within a security mechanism is adopted, the rigour will increase to QAM_RIG.2 if the actual means of verification is manual or to QAM_RIG.3 if the means of verification provides a lower latency in the verification of a software tool for this purpose.
Independence of verification is defined as the intersection between the set of individuals who were involved in the deployment of the security mechanism and those undertaking the verification. The capability levels for this metric are: QAM_IND.1, which will be assigned in the case of self-assessment, when the verification of the security mechanism is performed by individuals who set up the deployment or with probes. In contrast, QAM_IND.2 involves partial verification with independent means: the verification is partly performed by those who set up the security mechanism deployment features, while other properties are assessed by a third-party (external to the organisation or internal provided they did not take part in the deployment). The highest capability, QAM_IND.3, is assigned when all the functionalities and properties of the security mechanism are verified by a third-party.

The authors provide the following table, which shows an analysis of how these family capability levels are combined in order to arrive at a quality level for security verification. The matrix indicates that in order for a verification process to be at, for example, a level of quality of 3, at least the following requirements should be satisfied: QAM_COV.2, QAM_RIG.2, QAM_DPT.2 and QAM_IND.2.

Table 4-5. Capability levels combined for security verification

<table>
<thead>
<tr>
<th>Class</th>
<th>Quality family and meaning</th>
<th>Quality level QL</th>
</tr>
</thead>
<tbody>
<tr>
<td>QAM: security verification process</td>
<td>QAM_COV: coverage. Larger coverage of the verified security mechanism provides more confidence on the results about its status</td>
<td>1 2 2 2 3</td>
</tr>
<tr>
<td>verification quality metrics</td>
<td>QAM_DPT: depth. A detailed verification of the security mechanism will decrease the likelihood of undiscovered errors</td>
<td>1 2 2 3 4</td>
</tr>
<tr>
<td></td>
<td>QAM_RIG: rigour. The more structured the evaluation of the deployed security mechanism, the more reliable the outcome of the verification</td>
<td>1 2 2 3 3</td>
</tr>
<tr>
<td></td>
<td>QAM_IND: independence of verification. Verification performed by a third-party evaluator or a software tool provides more assurance</td>
<td>1 1 2 2 3</td>
</tr>
</tbody>
</table>

The task has two steps: “Application of evaluation” and “Analysis of evaluation”. This task will be performed using the transformation described in 5.3.6.

Figure 4-38 shows the Inputs, Outputs and steps of Task 5.1 according to SPEM 2.0.
The main stakeholders that are involved in this activity are software engineers, business managers, cloud experts, security experts and auditors.

As a result of this task, we will obtain a new version of the SMiLe2Cloud Model that will contain the metrics.

The techniques and practices required for the realisation of this task will be meetings and interviews with the people involved in the project.

Figure 4-39 shows a detailed description of Task 5.1 using SPEM 2.0.

**TaskUse: 5.1 Evaluation**

- **ProcessPerformer** {kind: primary}
  - RoleUse: **Business manager** {kind: in}
  - RoleUse: **Audit** {kind: in}
  - RoleUse: **Security Expert** {kind: in}
  - RoleUse: **Cloud Expert** {kind: in}
  - RoleUse: **Software engineer** {kind: in}
- **WorkDefinitionParameter** {kind: in}
  - WorkProductUse: **Services**
  - WorkProductUse: **SMiLe2Cloud Model**
  - WorkDefinitionParameter {kind: out}
  - WorkProductUse: **SMiLe2Cloud Model**
  - WorkProductUse: **Metrics**

- **Steps**
  - Step: **Application of evaluation**
  - Step: **Analysis of evaluation**

- **Guidance**
  - Guidance {kind: Practice}: **Meetings**
  - Guidance {kind: Practice}: **Interviews**
4.3.5.2. **Task 5.2 Definition of new requirements**

The cloud is a changing environment. Some of the issues that most experts are now studying were still undetected only a couple of years ago. In two years’ time, there might be completely new services that will help to strengthen the security of a LIS system migrated to the cloud. Furthermore, since we have delegated the responsibility for some controls, a continuous watch on the levels and metrics of security is advisable.

Even when the system is operating in working conditions, and as we have already seen, the verification of some parts requires a continuous effort in order to gather further evidence that the security is maintained at the levels agreed on and that security services are provided.

Activity 5.2 must, therefore, be periodically repeated, and the results must be analysed within the limits of the specifications of the security architecture proposed.

But even if the security specifications are met as written, basing our process on a Deming cycle signifies some sort of continuous re-evaluation of possible improvements to the system.

The improvements may originate from technical advances in the field, from changes in the SLA standard or the services that the cloud provider offers, from legislative grounds, etc.

The new requirements will be implemented as an XML file based on the SMiLe model. This file will be the input for activity 2, i.e. the analysis activity.

The task has two steps: “identify new requirements” and “definition of new requirements”. Figure 4-40 shows the Inputs, Outputs and steps of Task 5.2 according to SPEM 2.0.

![Figure 4-40. SPEM 2.0 view of Inputs, Outputs and steps of Task 5.2](image-url)

The main stakeholders that are involved in this activity are requirements engineers, software engineers, business managers, cloud experts and security experts.
As a result of this task, we will obtain a new version of the SMiLe2Cloud Model that will contain the definition of new requirements or goals.

SMiLe2Cloud is an iterative process. These new requirements will be the input for a new cycle in the SMiLe2Cloud framework. The new cycle starts with the Analysis Activity.

The techniques and practices required for the realisation of this task will be meetings and interviews with the people involved in the project.

Figure 4-41 shows the detailed description of Task 5.2 using SPEM 2.0.

```
TaskUse: 5.2 Definition of new requirements
  ProcessPerformer {kind: primary}
    RoleUse: Business manager {kind: in}
    RoleUse: Security Expert {kind: in}
    RoleUse: Cloud Expert {kind: in}
    RoleUse: Software engineer {kind: in}
    RoleUse: Requirements engineer {kind: in}
  WorkDefinitionParameter {kind: in}
    WorkProductUse: Metrics
    WorkProductUse: SMiLe2Cloud Model
    WorkDefinitionParameter {kind: out}
    WorkProductUse: SMiLe2Cloud Model
    WorkProductUse: Definition of new requirements
  Steps
    Step: Identify of new requirements
    Step: Definition of new requirements
  Guidance
    Guidance {kind: Practice}: Meetings
    Guidance {kind: Practice}: Interviews
```

Figure 4-41. Detailed description of Task 5.2 using SPEM 2.0.
5. SMiLe2Cloud Ecosystem
This chapter provides a description of an ecosystem of tools that can be used to support the migration process from legacy systems to cloud computing. This ecosystem, called as SMiLe2Cloud ecosystem, allows the use of the SMiLe2Cloud framework in all its activities.

The chapter is structured as follows. It begins with an introduction to the solution adopted, after which the technological environment employed for its development is described. A description of the tools comprising the SMiLe2Cloud ecosystem is then provided, and the chapter closes with a presentation of various contributions made by the tool and future work that will be carried out on it.

5.1. Introduction

The previous chapter provides a detailed description of SMiLe2Cloud, along with all the principal aspects and characteristics of the migration cycle, from the legacy system to its implementation in cloud computing. The range and reach of this thesis as regards constructing a new tool that will support the entire process appears to be an impossible task.

However, since the SMiLe2Cloud framework was designed on the basis of standard models and languages, it is possible to look for certain tools that enable certain tasks in the process to be managed semi-automatically.

After studying some of the available tools, in this chapter we compile those that may provide the process with the greatest value. Some of these tools can be reused, others must be adapted and most important, a new tool (the SMiLe2Cloud Ecosystem Tool) must be developed to integrate them.

The major contribution of the solution presented is the integrated and coordinated use of these tools to obtain adequate support for the proposed methodology. Each tool in itself can be used in only one of the activities in the process. The solution proposed, however, enables their integrated use in all the activities in the SMiLe2Cloud framework, from the extraction activity to the evaluation activity.

When describing the relationship among the tools presented in this chapter, it is not our intention to exclude other techniques or instruments, in the sense that any organisation can opt to use any other tool that it prefers, always supposing that it can be integrated into the models proposed.

5.2. Technological Environment

This thesis has been developed with the collaboration of researchers from the GSyA and SenSe research groups. The GSyA group is located in Ciudad Real (Spain), while the SenSe group is in Brighton (UK). This geographical distance could have been an obstacle to the development of the solution and the exchange of information in general. However, this problem was solved by employing Amazon Workspaces. Amazon WorkSpaces is a completely administered Desktop as a Service (DaaS) solution that is executed in Amazon Web Services.

The present thesis was developed by using the Microsoft Windows virtual desktop, which is based on the Amazon cloud. This made it possible to collaborate on the development of the solution presented in this chapter, thus enabling distributed access to the documents, applications and resources required.
An image of the virtual desktop shared by both research groups is shown in Figure 5-1.

![Virtual desktop shared](image)

**Figure 5-1. Virtual desktop shared**

### 5.3. **SMiLe2Cloud Ecosystem**

As mentioned previously, the proposed solution covers the entire migration process from legacy systems to cloud computing by following the SMiLe2Cloud framework.

The SMiLe2Cloud framework comprises 5 activities: extraction, analysis, design, deployment and evaluation. One or various tools to follow the SMiLe2Cloud process have been selected or adapted for each activity. The MARBLE and IBUPROFEN tools are, for example, used in the extraction activity, while the SecTro and STARWatch are used in the design and deployment activities, respectively. All of these tools will be explained in the following sections.

The tools were integrated by creating the SMiLe2Cloud Ecosystem, which enables the semi-automation of the SMiLe2Cloud process, thus facilitating, managing, validating and transforming the inputs and outputs of the various activities in the process.

A summary of the SMiLe2Cloud Ecosystem is provided in Figure 5-2.
5.3.1. **SMiLe2Cloud Ecosystem Overview**

The SMiLe2Cloud Ecosystem is a set of tools configured to work in an integrated way to support all the activities of SMiLe2Cloud framework. It is based on Pentaho Data Integration, which is an engine that allows the user to define data-integration and transformation tasks.

As will be noted in Figure 5-3, a general process has been created in Pentaho that includes the entire process defined in SMiLe2Cloud. This process (or job using the pentaho nomenclature\(^1\)) contains both the calls to the various applications and the execution of the various transformation tasks that allow the process to be automated.

This general process shown in the Figure 5-3 first calls the MARBLE and IBUPROFEN tools, after which a Process-to-goal transformation occurs, which transforms the outputs of the tools described into inputs for SecTro. This is followed by the align Security Mechanism phase, which introduces the security mechanisms into the set of metrics that will be evaluated during the Evaluation phase. The STARWatch tool is then used to carry out the Cloud Provider Selection and Deployment Specification phases. Finally, the Evaluation phase is produced, which obtains new requirements that will serve as the input for a new process cycle.

---

\(^1\) Following the pentaho nomenclature, a job is a set of transformations, jobs and steps that run sequentially. A transformation is a set of steps interconnected that process records from a data source and whose output is one or more registers.
As mentioned previously, the process is guided by the SMiLe2Cloud model. This has been implemented by defining an XML schema with its principal elements.

In the following sections the selected tools and the necessary adaptation will be detailed for each activity of the SMiLe2Cloud methodology.

5.3.2. Activity 1 Extraction

The extraction activity (see Section 4.3.1) consists of three tasks: “1.1 Define model on BPMN”, “1.2 Refactor model on BPMN” and “1.3 Define systems requirements”. The first task is carried out using the MARBLE tool and the second using the IBUPROFEN tool, both of which were developed by the Alarcos research group. The process-to-goal transformation of the SMiLe2Cloud Ecosystem is used for the third task.

The main characteristics of each tool used in each of the tasks of which the Extraction activity is composed are explained as follows.

5.3.2.1. Task 1.1: Define model on BPMN

The main objective of the task is to extract BPMN process models from the source code of the legacy system. MARBLE will be used in this activity because it supports business process archaeology by retrieving business processes from legacy source code (see section 3.3.1.5). MARBLE provide a first version of business process models that represents a more efficient solution and a good starting point from which to achieve a business process archaeology.

The MARBLE Tool has been developed as a plug-in in the Eclipse environment. The user interface of MARBLE Tool is shown in Figure 5-4:
5.3.2.2. **Task 1.2: Refactor model on BPMN**

The task refactors the extracted process model obtained in the previous task.

It introduces a set of ten refactoring algorithms which can be applied to business process models represented by means of graphs, expressed in BPMN. The IBUPROFEN Tool has been developed as a plug-in in the Eclipse environment. The user interface of IBUPROFEN Tool is shown in Figure 5-6:
5.3.2.3. **Task 1.3: Define systems requirements**

The task executes a process-to-goal transformation in order to create an initial goal model from the refactored BPMN process model.

This process-to-goal transformation will create an additional higher level of abstraction, represented by a goal model of the legacy system. The proposed transformation will essentially derive a goal model, in which security will be produced and expressed using Secure Tropos.

Transformation rules have been defined using concept mappings between Secure Tropos and BPMN concepts, and instructions on how they can be utilised are provided. A transformation algorithm has been defined in section 4.3.1.3.

This process has been automated by creating a transformation in the SMiLe2Cloud Ecosystem. The process-to-goal transformation is shown in Figure 5-6:

![Figure 5-6. SMiLe2Cloud Ecosystem process-to-goal transformation](image)

5.3.3. **Activity 2 Analysis**

This activity is composed of two tasks: “2.1 Analysis of Security Requirements” and “2.2 Align Security Mechanism”. The first task is carried out using the SecTro tool, which was developed by the SEnSe research group, while the second task is carried out using the SMiLe2Cloud Ecosystem.

5.3.3.1. **Task 2.1: Analysis of Security Requirements**

The analysis activity is a crucial part of the SMiLe2Cloud process as the security requirements are extracted and defined on the basis of the system requirements elicited from the previous activity. The input for this activity is the system requirements elicited during the previous extraction activity. It is based on the BPMN model and is converted into the SMiLe2Cloud model as an Extensible Markup Language (XML) file.

The system requirements are modelled using existing elements, such as goal, actor, plan, resource and constraints in Secure Tropos, in order to visually represent the relationships and requirements of the legacy system. We then extend the Secure Tropos notation to capture the security requirements for cloud systems and define secure services, which are analysed in the proposed cloud infrastructure view so as to model components in both the application and physical layers.

SecTro is a standalone application that was built with the Java programming language, signifying that this application is portable across different platforms.
SecTro’s workspace consists of the drawing canvas in the centre, a series of tabs to show the diagrams developed for each stage of Secure Tropos at the top, the project explorer and the properties panel on the right-hand side and the toolbox on the left-hand side. The SecTro assistant is at the bottom of the workspace (see Figure 5-7).

The main functionalities of the SecTro are to support the developer when modelling the Secure Tropos activities. The tool, therefore, enables the developer to perform security reference modelling, security constraint modelling, secure entities modelling, and secure capability modelling. During these activities, the tool has a mechanism with which to check the rules and constraints, and informs the developer about any errors. The SecTro assistant panel also shows more information about the rules and constraints, the concepts and the meta-models. It thus assists the developer with the learning process of the Secure Tropos methodology and additionally enables the developer to export the diagrams as images and in XML format.

![Figure 5-7. SecTro](image)

5.3.3.2. **Task 2.2: Align Security Mechanism**

The objective of this task is to align security mechanisms and cloud controls. The security goal dimensions and the security counter measure dimensions are identified for each security mechanism. The cloud controls are aligned with the security mechanisms by means of security goal and security countermeasure dimensions. There are some related controls for each security goal and security countermeasure dimension.

An outline of the list of security controls of CCM v3 aligned with the RMIAS security goal dimensions and security countermeasure dimensions is provided in the table in Annex II.

This process has been automated by creating a transformation in the SMiLe2Cloud Ecosystem. The input for this transformation is the SMiLe2Cloud Model that contains the security mechanisms within the security goal dimensions and the security countermeasure dimensions. This transformation accesses the database described in Annex II. The
transformation then executes an algorithm in JavaScript that makes the alignment required to obtain the cloud controls. The output is the SMiLe2Cloud Model that contains the cloud controls.

The alignment between security mechanisms and cloud controls is shown in Figure 5-8:

![SMiLe2Cloud Model alignment](image)

5.3.4. **Activity 3 Design**

The design activity is focused on selecting the service model from the implementation model and making the decision regarding which cloud provider to select on the basis of the STAR certificate (Cloud Security Alliance, 2014). The input is the specification of the security requirements obtained in the previous activity. The requirements specification is aligned with CCM v3 in order to make available a list of controls that our cloud provider should fulfil in order to attain the level of security required.

The design activity consists of two tasks, “3.1 Identification of Deployment and Service Model” and “3.2 Selection of cloud provider”. This activity is carried out using the STAR Watch tool provided by the Cloud Security Alliance (CSA).

5.3.4.1. **Task 3.1: Identification of Deployment and Service Model**

This task is in charge of identifying the implementation and service model to which we wish to migrate, bearing in mind factors such as the customer’s needs, the fulfilment of requirements, the resources available and the level of security that we wish to obtain in the new cloud system to which we shall migrate. This task is carried out using the CSA STARWatch, which helps and facilitates the identification of the most appropriate model for our migration.

CSA STARWatch (see Figure 5-10, a Compliance and Assurance as a Service (CAaaS) application) helps organizations better visualize and manage the compliance of the CSA STAR database and requirements.
STARWatch employs the CSA’s proprietary Cloud Control Matrix (CCM) and eliminates the need for complex and lengthy spreadsheets used to gather information and assess compliance across a variety of standards.

CSA STARWatch provides customers with rapid responses to their compliance questions, in order to ensure a common baseline for security compliance and to drive efficiency with the integration of cloud services for more than 200 cloud service providers.

5.3.4.2. Task 3.2: Selection of cloud provider

Once the implementation model and the service model have been selected, and the security requirements of the system that are aligned with the CSA domains have been made available, it is possible to select the cloud provider that best fits the security needs according to the STAR standard. The STAR standard provides, for each cloud provider, the list of controls implemented, which are obtained from those defined in the CCM v3 matrix.

Since the list of controls needed to cover the security requirements defined in the analysis activity (in the SMiLe2Cloud model) is available, as is the list of cloud providers that fulfil these controls (thanks to STARWatch), it is possible to identify the list of cloud providers that fulfil the security requirements.

This process has been automated by creating a transformation in the SMiLe2Cloud Ecosystem. The input is the SMiLe2Cloud Model that contains the cloud controls and the service and deployment model. This transformation accesses the STARWatch database in order to obtain all the data from the main cloud providers. The job then executes an algorithm in JavaScript that creates the alignment required to obtain the cloud service providers that match with the controls, the service and the deployment model. The output is a list of cloud providers that fulfil all the security requirements.

The selection of a cloud provider is shown in Figure 5-11:
Once the list of cloud providers that fulfill the security requirements has been obtained, it is necessary to select one of them according to other variables, such as cost, the technologies to be implemented, etc. This is done by carrying out a SWOT analysis.

5.3.5. **Activity 4 Deployment**

This activity is composed of two tasks: “4.1 Deployment Specification” and “4.2 Implementation”. It is carried out using the SMiLe2Cloud Ecosystem.

5.3.5.1. **Task 4.1: Deployment specification**

Once we have defined the security controls that our application must meet and we have the cloud provider to which we are migrating, the STAR standard will guide us during the implementation.

As mentioned previously, CSA Star Watch contains, for each cloud provider, the list of controls it implements, as obtained from those defined in the CSA’s Cloud Controls Matrix (CCM) (Cloud Security Alliance, 2016a).

This process has been automated by creating a transformation in the SMiLe2Cloud Ecosystem. The input is the SMiLe2Cloud Model that contains the cloud controls and the cloud provider selected. This job accesses the Star Watch database in order to obtain all the data from the cloud provider selected.

This information makes it possible to identify the security services of the provider selected. The output is the deployment specification.

The identification of security services is shown in Figure 5-12:
5.3.5.2. **Task 4.2: Implementation**

Finally, the implementation itself takes place. During the implementation task it may be necessary to contract the services and to sign the Service Level Agreement (SLA), develop the custom security elements or set all the security controls in working condition.

The implementation will depend on the cloud provider chosen.

5.3.6. **Activity 5 Evaluation**

Once the entire process has been moved to the cloud in a secure manner, it is time to validate the security of the system.

This activity is composed of two tasks: “5.1 Evaluation” and “5.2 Definition of new requirements”. This activity has been carried out by creating a transformation in the SMiLe2Cloud Ecosystem. This transformation makes it possible to access the database of the metrics and to carry out the calculations required to validate the process. The outputs are the new requirements that will be the input for the SecTro tool.

The evaluation of the system is shown in Figure 5-13:
5.4. **Contributions of the Tools and Future Work**

This chapter provides descriptions of the various tools employed to automate and facilitate the execution of the migration process of legacy applications to cloud computing by means of the SMiLe2Cloud framework. This set of tools has made it possible to fulfil the objectives initially proposed, i.e. facilitating the application of the principal proposal of this doctoral thesis.

The main contribution of the solutions presented is the integration of diverse tools, thus allowing the set as a whole to guide the migration process defined in the SMiLe2Cloud framework. Each tool in itself can be used only in one of the activities in the process. However, the solution proposed allows the use of the SMiLe2Cloud framework in all its activities, from the extraction activity to the evaluation activity.

Part of our future work as regards the SMiLe2Cloud support tools concerns extending the tools used in the extraction activity. The tools chosen allow only certain programming languages to be extracted. Extending these tools to other programming languages in which the legacy systems are developed could extend the use of the methodology.

Finally, a much more ambitious objective than that stated previously would be to develop a tool that is capable of completely supporting the SMiLe2Cloud framework as a whole. This development could be carried out by using the requirements and characteristics identified in the best individual tools chosen in the previous step as a basis.
6. Case study
This chapter shows how the SMiLe2Cloud framework presented in Chapter 4 was applied in the two different case studies using the Action-Research method as detailed in Chapter 2. First, in Section 6.1, we provide a brief introduction to the Organisation in which SMiLe2Cloud was applied, after which the first and second case studies are described in Sections 6.2 and 6.3, respectively. Finally, a brief reflection on the lessons learned from the case studies is shown in Section 6.4.

6.1. Introduction

Both of the case studies presented were carried out at the “Comisión Nacional de los Mercados y la Competencia” (CNMC – the National Commission for Markets and Competition), an organisation that promotes and defends the smooth operation of all markets in the interest of consumers and companies in Spain.

The CNMC was created in 2013 from the amalgamation of six organisations: Comisión Nacional de la Competencia (CNC – National Commission for Competition), Comisión Nacional de Energía (CNE – National Energy Commission), Comisión del Mercado de las Telecomunicaciones (CMT – National Telecommunications Market Commission), Comisión Nacional del Sector Postal (CNSP – National Postal Sector Commission), Consejo Estatal de Medios Audiovisuales (State Council for Audio-visual media) and Comité de Regulación Ferroviaria y Aeroportuaria (Committee for the Regulation of Railways and Airports).

This amalgamation, in addition to the fact that each organisation had its own highly diverse technologies, signified that it was necessary to migrate the different information systems.

One of these systems is the Registro de Entrada Masivo (Massive Input Registry, from here on, REM), a system that was developed by the former CNE that enables the transmission of large amounts of data in a confidential and integral manner. Another of the systems is WECO, the system developed by the former CHC that allows the management of files in accordance with Law 39/2015.

The practical application of SMiLe2Cloud to these two systems, REM and WECO, which it was necessary to migrate to the cloud, is described in the following sections.

6.2. REM

The Registro de Entrada Masivo (REM) is the application used by the CNMC to manage the exchange of large amounts of data between the organisation and its customers. The system must comply with a series of security requirements owing the sensitive nature of the data that it handles.

Section 6.2 is structured as follows: First, in Section 6.2.1, we provide a brief description of the system in which SMiLe2Cloud has been applied and the problem that had to be solved, after which, in Section 6.2.2, we present the application of the SMiLe2Cloud activities along with the application of the SMiLe2Cloud-Ecosystem tool developed to support those activities, which were explained in Chapter 5.
6.2.1. **Description of the System**

The REM is a service that the CNMC offers companies in order to send information in a secure telematic manner. As the data that are exchanged are subject to the regulations contained in the Organic Law for Data Protection (Ley Orgánica de Protección de Datos – LOPD), the system must comply with a series of security requirements:

- **Authentication:** the users are who they say they are.
- **Integrity:** the data are not manipulated, and are stored in the same way that they were sent by the customer.
- **Non-repudiation:** no user can deny having received or sent data.
- **Secure Exchange:** the data are exchanged via a secure channel and are correctly protected in the servers.

The system is accessed via a secure connection, that additionally requires the customer’s authentication. In order for different types of users to access REM they must, therefore, have a valid cryptographic certificate (that can be stored in either a cryptographic card or a file installed in their browser). Upon authenticating access, the system is, therefore, able to discover who the user who has entered the application is.

The customers can also upload files. This application carries out the transference by signing data authentication and document transference. The confidentiality of the data is thus guaranteed when transferring the documents via a coded channel (SSLv3/TLS1.0).

Once the transference has been completed, it is possible to download a pdf file containing the data delivered.

6.2.1.1. **Functional Description**

The functional requirements of the system, as shown in Figure 6-1, are listed below:

- **Send ‘large’ documents:** the system should support the sending of documents of an ‘unlimited’ size.
- **Session Control:** the system should provide a client that is able to control and follow up on unfinished sessions, along with its deletion.
- **Interrupt and postpone messages sent:** it should be possible to interrupt the transference. This does not mean that its state is deleted from the system, and the idea is that it can be restarted.
- **Resume postponed transference:** after interrupting the sending of a message, it should be possible to restart the process where it was left off, or at a nearby point. It should be borne in mind that these are large files and it is not desirable to have to start again from the beginning of an interrupted transference.
- **Delete postponed transference:** it is sometimes desirable to delete an unfinished transference, such as a mistake in the file to be transferred. It should, therefore, be possible to interrupt it and then delete it.
Faults tolerance in the Web: it is necessary to take the measures required to ensure that a transference is not corrupted such that it has to be reinitiated. These measures should allow the transference to remain unfinished, but in no case should be deleted.

6.2.1.2. Architecture

As will be noted in Figure 6-2, the application consists of a reference client based on Java Web Start. This client incorporates certificate management mechanisms, the control and follow up of sessions and an intuitive graphic interface that is used to carry out all the operations.
6.2.1.3. **Interface**

The interface that controls the transference is shown in Figure 6-3. The information regarding the session is shown divided into two groups:

- The first contains the sender’s data (personal data, certificate, distributor, etc.).
- The second group shows the data concerning the transference and its state on a progress bar.

![Figure 6-3. REM transmission model interface](image)

6.2.2. **Application of SMiLe2Cloud**

This section provides a description of the application of SMiLeCloud in order to migrate the REM system to the cloud. Each of the activities will be analysed, and the principal input and output flows specified.
6.2.2.1. **Extraction Activity**

The Extraction activity has three tasks: extraction of BPMN process models from the source code of the legacy system, refactoring of the process model extracted and a process-to-goal transformation to create an initial goal model from the refactored BPMN process model.

The result will be a business process model, aligned with the organisation’s high-level objectives. This process model operationalises the system requirements captured at the goal model level. The SMiLe2Cloud Model includes this system requirements goal model. This includes goals, actors, plan, resources and constraints.

6.2.2.1.1. **Tasks 1.1 and 1.2: Application of MARBLE and IBUPROFEN to the case study. Obtaining the BPMN**

The first step requires the legacy process supported by this system to be transformed into a goal model in which the elicitation of the security requirements will take place. MARBLE + IBUPROFEN will be utilised to extract a process model for the source code of the legacy system.

In this example, the main module of the REM system is selected for further analysis. Since this module, which represents the system’s home screen, provides links to the rest of the system's main functionalities, it allows a complete overview of the REM system. Parts of the code of the main module (VentanaPrincipal.java) are shown in the Figure 6-5:
The legacy process model produced by the MARBLE+IBUPROFEN tool after some minor readjustments is presented in the Figure 6-6:

The process begins with a session initialisation during which the user is connected to the system via the REM client, followed by the establishment of a secure connection between the user and the server. The file transfer occurs as an iterative task during which the user can upload or download files split into smaller parts. Each transfer of or modification made to the files is registered in a transfer log, which can also be downloaded by the user at the end of each session. Finally, the user has the option of either restarting a transfer session or deleting the existing session, thus stopping the connection between the client and the server.

The output is the generation of an XML document by the BPMN model, as is shown in the following figure:
6.2.2.1.2. Task 1.3: Transformation of BPMN into Goal Model

The application of the transformation rules defined in Table 4-3 of the section 4.3.1.3 to the legacy process shown in Figure 6-6 enables a SMiLe2Cloud goal model to be produced, as is illustrated in Figure 6-8.
In this goal model, the actor denoted as REM represents the system, while each of the goals represents an activity in the legacy process, with the top-goal being added manually in order to encapsulate the overall purpose of the REM system’s functionality.

The following goals have been identified: facilitate large file transfers, set up connection, finalize session, send part, restart session, update transfer logs, delete session, download transfer logs, sign transfer logs, initialize session, set session ID, set token ID, add custom certificates and CRLs and check security certificates. Furthermore, a resource (transfer logs) has been identified.

The output of this activity is the generation of an XML document by the SMiLe2Cloud goal model, as is shown in the Figure 6-9:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE ADOMML SYSTEM "adomdl31.dtd"/>
<ADOMML version="3.1" date="26.11.2015" time="23:59" database="adomdb" username="Admin" adoversion="Version 1.8 4.0"/>
<MODEL>
  <MODEL id="mod.11523" name="REM-legacy" version="" modeltype="Security Model" l1type="bp" applib="SecTro2_stable.git-1c3e053">
    <MODELATTRIBUTES>
      <ATTRIBUTE name="Keywords" type="STRING"></ATTRIBUTE>
      <ATTRIBUTE name="Description" type="STRING"></ATTRIBUTE>
      <ATTRIBUTE name="Comment" type="STRING"></ATTRIBUTE>
      <ATTRIBUTE name="Model" type="ENUMERATION">
        <CURRENTMODEL>
        </CURRENTMODEL>
      </ATTRIBUTE>
      <ATTRIBUTE name="State" type="ENUMERATION">
        <INPROGRESS></INPROGRESS>
      </ATTRIBUTE>
      <ATTRIBUTE name="Reviewed on" type="STRING"></ATTRIBUTE>
      <ATTRIBUTE name="Reviewed by" type="STRING"></ATTRIBUTE>
      <ATTRIBUTE name="Author" type="STRING">Admin</ATTRIBUTE>
      <ATTRIBUTE name="Creation date" type="STRING">18.11.2015, 12:18</ATTRIBUTE>
      <ATTRIBUTE name="Last user" type="STRING">Admin</ATTRIBUTE>
      <ATTRIBUTE name="Date last changed" type="STRING">20.11.2015, 23:58:28</ATTRIBUTE>
      <ATTRIBUTE name="Number of objects and relations" type="INTEGER">0</ATTRIBUTE>
    </MODELATTRIBUTES>
  </MODEL>
</ADOMML>
```

Figure 6-9. REM SMiLe2Cloud goal model in XML

### 6.2.2.2. Analysis Activity

Security requirements are mainly obtained by analysing an organisation’s attitude towards security and after studying its security policy. The security requirements are modelled as security constraints and a set of security objectives and a set of security mechanisms are identified for each one of them.

This activity is composed of two tasks: Analysis of Security Requirements and Align Security Mechanisms.

#### 6.2.2.1. Task 2.1: Analysis of Security Requirements

In the “Analysis of Security Requirements” task, the SMiLe2Cloud Model is used to derive a set of security requirements with which the system must comply in the new environment. Security requirements are mainly obtained by analysing an organisation’s attitude towards security and after studying its security policy.

The analysis of security requirements is a crucial part of the SMiLe2Cloud process as the security requirements are extracted and defined on the basis of the system requirements elicited from the previous activity. The input for this activity is the system requirements elicited during the previous extraction activity, which is based on the BPMN model and converted into the SMiLe2Cloud model.
During this task, security constraints, threats, security objectives and security mechanisms are identified and introduced into the goal SMiLe2Cloud model.

The SMiLe2Cloud model, therefore, includes, as an output of this task, the system security requirements, which include threats, security constraints, security objectives and security mechanisms.

This task is performed using the SecTro tool, as shown in Figure 6-10:

![Figure 6-10. SecTro tool](image)

The result obtained after applying the security requirements analysis task to the case study is shown in the Figure 6-11. Two threats have been identified (eavesdropping and user impersonation). Moreover, five security constraints have been found: access only to authorised users, customisation of security measures must be allowed, transferred data must remain confidential, data must not be altered during transfer and transfer of data must not be repudiated. In addition to this, the following security mechanisms have been identified: user whitelist, login with digital certificate, custom certificates, advanced certificate signature frameworks, custom network settings, encrypted channel, verification token and digital signature.
6.2.2.2. **Task 2.2: Align Security Requirements with Cloud Security Controls**

The objective of this task is to map the security mechanism identified in the previous step onto the cloud controls specified in the Cloud Control Matrix - CCM.

The security goals and the security countermeasures are identified for each security mechanism. The cloud controls are aligned with the security mechanisms by means of security goals and security countermeasures. There are some controls related to each security goal and security countermeasure.

The output of this task in the SMiLe2Cloud Model is that the system security requirements are aligned with CSA Domains. This includes security goals, security countermeasures and cloud controls.
For example, the following security goals have been identified for the security mechanism “encrypted channel”: confidentiality, integrity, non-repudiation. The technical countermeasure dimension has also been identified.

The “Align security mechanism” algorithm described in section 4.3.2.2.3 is used to obtain the list of controls that apply to the security goals identified (confidentiality, integrity and non-repudiation) and to the technical countermeasure dimension shown in Figure 6-12: AIS-03 - Data Integrity, AIS-04 - Application & Interface Security Data Security, EKM-01 – Entitlement, EKM-02 – Key Generation, EKM-04 - Storage and Access, IVS-06 - Network Security, IVS-10 – VM Security - Data Protection and MOS-11 – Encryption.

![Figure 6-12. REM SMiLe2Cloud cloud controls](image)

The same process was carried out for the various security mechanisms: user whitelist, login with digital certificate, custom certificates, advanced certificate signature frameworks, custom network settings, verification token and digital signature.

The list of controls that are applicable to these security mechanisms once this task has been applied to all the security mechanisms identified is shown in Table 6-1.
<table>
<thead>
<tr>
<th>Domain</th>
<th>Control</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application &amp; Interface Security</td>
<td>Data Integrity</td>
<td>AIS-03</td>
</tr>
<tr>
<td></td>
<td>Data Security/Integrity</td>
<td>AIS-04</td>
</tr>
<tr>
<td>Data Security &amp; Information Lifecycle</td>
<td>Controlled Access Points</td>
<td>DCS-02</td>
</tr>
<tr>
<td>Management</td>
<td>Identification</td>
<td>DCS-03</td>
</tr>
<tr>
<td></td>
<td>Unauthorised Persons Entry</td>
<td>DCS-08</td>
</tr>
<tr>
<td>Datacentre Security</td>
<td>Entitlement</td>
<td>EKM-01</td>
</tr>
<tr>
<td></td>
<td>Key Generation</td>
<td>EKM-02</td>
</tr>
<tr>
<td></td>
<td>Storage and Access</td>
<td>EKM-04</td>
</tr>
<tr>
<td>Security Identity &amp; Access Management</td>
<td>Credential Lifecycle / Provision Management</td>
<td>IAM-02</td>
</tr>
<tr>
<td></td>
<td>Diagnostic / Configuration Ports Access</td>
<td>IAM-03</td>
</tr>
<tr>
<td></td>
<td>Segregation of Duties</td>
<td>IAM-05</td>
</tr>
<tr>
<td></td>
<td>User Access Authorisation</td>
<td>IAM-09</td>
</tr>
<tr>
<td></td>
<td>User Access Revocation</td>
<td>IAM-11</td>
</tr>
<tr>
<td></td>
<td>User ID Credentials</td>
<td>IAM-12</td>
</tr>
<tr>
<td></td>
<td>Utility Programmes Access</td>
<td>IAM-13</td>
</tr>
<tr>
<td>Infrastructure &amp; Virtualisation</td>
<td>Network Security</td>
<td>IVS-06</td>
</tr>
<tr>
<td></td>
<td>OS Hardening and Base Controls</td>
<td>IVS-07</td>
</tr>
<tr>
<td></td>
<td>Production / Non-Production Environments</td>
<td>IVS-08</td>
</tr>
<tr>
<td></td>
<td>VM Security - Data Protection</td>
<td>IVS-10</td>
</tr>
<tr>
<td></td>
<td>Hypervisor Hardening</td>
<td>IVS-11</td>
</tr>
<tr>
<td></td>
<td>Wireless Security</td>
<td>IVS-12</td>
</tr>
<tr>
<td></td>
<td>Network Architecture</td>
<td>IVS-13</td>
</tr>
<tr>
<td>Interoperability &amp; Portability</td>
<td>Virtualisation</td>
<td>IPY-05</td>
</tr>
<tr>
<td>Mobile Security</td>
<td>Cloud Based Services</td>
<td>MOS-06</td>
</tr>
<tr>
<td></td>
<td>Device Management</td>
<td>MOS-10</td>
</tr>
<tr>
<td></td>
<td>Encryption</td>
<td>MOS-11</td>
</tr>
<tr>
<td></td>
<td>Jailbreaking and Rooting</td>
<td>MOS-12</td>
</tr>
<tr>
<td></td>
<td>Lockout Screen</td>
<td>MOS-14</td>
</tr>
<tr>
<td></td>
<td>Remote Wipe</td>
<td>MOS-18</td>
</tr>
<tr>
<td></td>
<td>Security Patches</td>
<td>MOS-19</td>
</tr>
<tr>
<td></td>
<td>Users</td>
<td>MOS-20</td>
</tr>
<tr>
<td>Supply Chain Management, and Transparency</td>
<td>Data Quality and Integrity</td>
<td>STA-01</td>
</tr>
<tr>
<td>Accountability</td>
<td>Network / Infrastructure Services</td>
<td>STA-03</td>
</tr>
<tr>
<td>Threat &amp; Vulnerability Management</td>
<td>Anti-Virus / Malicious Software</td>
<td>TVM-01</td>
</tr>
<tr>
<td></td>
<td>Vulnerability / Patch Management</td>
<td>TVM-02</td>
</tr>
<tr>
<td></td>
<td>Mobile Code</td>
<td>TVM-03</td>
</tr>
</tbody>
</table>
6.2.2.3. Design Activity

The design activity is focused on selecting the service model and the deployment model depending on the customer’s needs, compliance with requirements, the resources available, etc. This is done using the distinction made by the NIST (National Institute of Standards and Technology) as a basis.

Once the deployment model and service model have been selected, and the system security requirements aligned with the CSA domains are available, the user can select the cloud provider that best fits the security needs using CSA Security, Trust & Assurance Registry (STAR) as a basis. The STAR standard provides, for each cloud provider, the list of controls that it implements.

The output of this activity included in the SMiLe2Cloud Model is the deployment model and the service model.

6.2.2.3.1. Task 3.1 Identification of Implementation model and service model

Identification of implementation model

As mentioned previously, the NIST distinguishes between the following implementation models: public cloud, private cloud, community cloud and hybrid cloud.

The REM application is a CNMC application. The CNMC does not, as an autonomous organisation, currently have a private cloud at its disposition. The CNMC depends on the Ministry of Economy and Competition which, in turn, depends on the General State Administration (AGE, Administración General del Estado). One of the AGE’s challenges in the near future is the CORA report (CORA, 2015), which proposes the creation of a private cloud that will provide the various national public administrations with a private cloud. This is, however, still at the project stage, signifying that the use of a private cloud, a community cloud or a hybrid cloud is not viable.

The only implementation model that is applicable in this case study is, therefore, the public cloud.

Identification of the service model

As mentioned previously, the NIST distinguishes between the following service models: Cloud Software as a Service (SaaS), Cloud Platform as a Service (PaaS) and Cloud Infrastructure as a Service (IaaS).

In the case of the Cloud Software as a Service (SaaS) model, the user can neither manage nor control the individual applications that are available as a service. The REM software is highly specific to the CNMC and the SaaS model cannot, therefore, be applied in our case study.

With regard to the Cloud Platform as a Service (PaaS) model, the user can neither manage nor control the underlying infrastructure, which includes the Web, the servers, the operative systems or the storage. The CNMC has highly qualified personnel for the management of the underlying infrastructure and it was, therefore, considered that the most appropriate service model for our case study would be the Cloud Infrastructure as a Service (IaaS) model.
The Cloud Infrastructure as a Service (IaaS) model provides processing, storage, an interconnection with the Web and other fundamental computational resources in which the user can install and execute arbitrary software that can include operative systems and applications. The user does not manage or control the underlying Cloud infrastructure, but does have control over the operative systems, the storage, the applications deployed and, possibly, a limited control over certain Web components.

6.2.2.3.2. Task 3.2 Selection of the Cloud provider

The input for the SMiLe2Cloud process design activity is based on the security requirement specification aligned with the CSA domains. The input point is, therefore, the list of controls defined in the Cloud Control Matrix v3, which are applicable to the case study described in Table 6-1.

Selection of cloud provider

Once the implementation model and the service model have been selected, and the security requirements of the system have been aligned with the CSA domains, it is possible to select the cloud provider that, according to the STAR standard, best fits with the security needs.

The STAR indicates the degree to which cloud providers comply with the controls defined in the CCM matrix, thus making it possible to identify those cloud providers that will guarantee the fulfillment of the system needs.

The SMiLe2Cloud tool facilitates this process. This tool communicates with the STARWatch application and obtains a database containing the various cloud providers and their degree of compliance with each CCM control. The list of providers that best fit in with the needs can thus be obtained automatically.

This case study is based on only two main cloud providers. According to the study carried out by Canalys (Canalys, 2017), the leaders in this field are AWS from Amazon and Azure from Microsoft. Between the two of them they have almost 50% of the market share, (Amazon AWS 31% and Microsoft Azure almost 16%).

The degree to which these providers comply with the controls defined in the analysis activity for a public cloud and the IaaS model, as defined in the previous task, is shown in Figure 6-13.
It can be concluded that both cloud providers are appropriate for the security management required. In order to choose between the two of them, we therefore carried out a study of their costs and a SWOT analysis for each one.

The costs were studied by bearing in mind the physical infrastructure of the REM application, which is detailed in the following table.

Table 6-2. Physical infrastructure of REM

<table>
<thead>
<tr>
<th>Infrastructure</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web Servers</td>
<td>1</td>
</tr>
<tr>
<td>Application Servers</td>
<td>1</td>
</tr>
<tr>
<td>Database Servers</td>
<td>1</td>
</tr>
<tr>
<td>Storage</td>
<td>1 TB</td>
</tr>
<tr>
<td>Backup</td>
<td>1 TB</td>
</tr>
<tr>
<td>VPN</td>
<td>1</td>
</tr>
</tbody>
</table>

The following configuration was chosen in order to simplify all the servers: Linux as an operative system, 8 core, 15 GB of RAM memory and SSD.

The comparison of the costs for Microsoft Azure and Amazon AWS are shown in the table below.

Table 6-3. Comparison of costs

<table>
<thead>
<tr>
<th>Infrastructure</th>
<th>Azure</th>
<th>AWS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Servers</td>
<td>585 €</td>
<td>681 €</td>
</tr>
<tr>
<td>Storage</td>
<td>91,87 €</td>
<td>107,6 €</td>
</tr>
<tr>
<td>Backup</td>
<td>201,07 €</td>
<td>40,9 €</td>
</tr>
<tr>
<td>VPN</td>
<td>152,74 €</td>
<td>96,9 €</td>
</tr>
<tr>
<td>Total</td>
<td>1.030,68 €</td>
<td>926,04 €</td>
</tr>
</tbody>
</table>

In order to complement the study of costs, we also carried out a SWOT analysis of both cloud providers in relation to our case study.

The SWOT analysis of Microsoft Azure and Amazon AWS with regard to REM is shown in the following figure.
The AWS cloud offers the most advantageous price for our case study. However, as the difference was only slight, we used the SWOT analysis in order to make the decision regarding which infrastructure to choose.

When considering the SWOT analysis, it is important to highlight that both Microsoft Azure and Amazon AWS are widely established on the market (with more than 55% of the market share, as stated previously). Amazon AWS (33%) and Microsoft (16%), but the latter is, according to Canalys (Canalys, 2017), growing at a greater rate.

Although both cloud providers are widely established on the market, Microsoft Azure’s weak integration with non-Microsoft technologies, along with its higher cost, signified that the supplier eventually chosen was cloud Amazon AWS.

6.2.2.4. Deployment Activity

The deployment activity is focused on developing the deployment specification and the implementation of the system.

6.2.2.4.1. Task 4.1 Deployment Specification

Once the list of applicable controls has been obtained and the service provider selected, it is time to identify the services of that provider that will guarantee the fulfilment of security.
As indicated in Task 4.1 “Deployment Specification”, the Consensus Assessments Initiative Questionnaire (CAIQ) can be used as a basis to identify the security services of the cloud provider chosen, in this case (CAIQ, 2017).

For example, according to the questionnaire used to guarantee the control “IAM-12 User ID Credentials”, it is necessary to use the Identity and Access Management (AWS IAM) service.

If the same is done for the remaining items on the list of controls identified, we obtain the list of services that it is necessary to implement in Amazon. This list should be refined by an expert. For example, if database characteristics are not used, it is possible to eliminate all those services related to databases from the list:

- AWS Credentials
  - Passwords
  - AWS Multi-Factor Authentication (AWS MFA)
  - Access Keys
  - Key Pairs
  - X.509 Certificates
  - Individual User Accounts
  - Secure HTTPS Access Points
  - Security Logs
  - AWS Trusted Advisor Security Checks
- AWS Service-Specific Security
  - Amazon Elastic Compute Cloud (Amazon EC2) Security
  - Networking Services
  - Amazon Virtual Private Cloud (Amazon VPC) Security
  - Amazon Route 53 Security
- Storage Services
  - Amazon Simple Storage Service (Amazon S3) Security
- Deployment and Management Services
  - AWS Identity and Access Management (AWS IAM)
  - Amazon CloudWatch Security
  - AWS CloudHSM Security

6.2.2.4.2. **Task 4.2 Implementation**

This case study was implemented on the Amazon cloud services.
Architecture

The migration of the application supposed a new development using modern technologies. The technologies used are described as follows:

- **JavaScript**: a programming language in the client sphere.
- **TypeScript**: a free open code programming language developed by Microsoft. It is a subset of JavaScript.
- **AngularJS**: an open code JavaScript framework, maintained by Google that is used to create and maintain Web applications on a single page. Its objective is to increase the applications by using the browser with a Modelo Vista Controlador (MVC) as a basis, in an effort to make the development and tests easier.
- **SystemJS**: a framework that is in charge of dynamically charging the js files.
- **JSPM**: package handling.
- **GULP**: a framework with which to construct the application. It helps to automate certain tasks, such as compressing or deploying code. ("Task running")
- **Bootstrap**: an HTML, CSS and JS framework for the responsive development of an application.
- **ui-bootstrap**: a framework from the Angular-UI family which adapts bootstrap to Angular via a set of directives.
- **ui-grid**: a framework from the Angular-UI family that provides a data-table from a json
- **Jasmine**: unitary test framework.
- **Protractor**: a e2e (end-to-end) framework designed by Angular.


Interface

The Figure 6-15 shows the Interface deployed in Amazon:
Infrastructure deployed

A capture of the infrastructure deployed in Amazon, and more specifically that related to the control panel of virtual private clouds (VPC), is shown in the Figure 6-16.

Service Level Agreements

Finally, it is necessary to sign service level agreements with the provider. In this case, the service level agreement signed contains:

- Supports the first level, which includes priority attention and telephonic support.
- 99.6% availability of SLA for AWS services.
• Customer service line 7x24x365
• Attention of 2 hours.

6.2.2.5. **Evaluation Activity**

The Evaluation activity has two tasks: the application of evaluation metrics and the definition of new requirements.

6.2.2.5.1. **Task 5.1 Application of evaluation metrics**

As explained in Section 4.3.5.1, there is a set of metrics with which to discover the level of quality of the security mechanisms selected for use and whether they are verifiable and can be audited, in order to discover the level of traceability of the security aspects and an assurance of the security that it is hoped that the product being migrated will attain.

As mentioned previously, (see Section 6.2.2.2.1), we have identified a set of security mechanisms for REM, which were obtained from the security requirements analysed. These are: user whitelist, login with digital certificate, custom certificates, advanced certificate signature frameworks, custom network settings, encrypted channel, verification token and digital signature.

Table 6-4. Security mechanism categories

<table>
<thead>
<tr>
<th>Category</th>
<th>Security Mechanism</th>
</tr>
</thead>
<tbody>
<tr>
<td>Secure Login</td>
<td>user whitelist</td>
</tr>
<tr>
<td></td>
<td>login with digital certificate</td>
</tr>
<tr>
<td>Certificate Management</td>
<td>custom certificates</td>
</tr>
<tr>
<td></td>
<td>advanced certificate signature frameworks</td>
</tr>
<tr>
<td></td>
<td>custom network settings</td>
</tr>
<tr>
<td></td>
<td>verification token and digital signature</td>
</tr>
<tr>
<td>Secure Channel</td>
<td>encrypted channel</td>
</tr>
</tbody>
</table>

• Secure Login. The authentication process should begin by obtaining the user’s ID number, his/her certificate and the ID of the Tomcat session coded with the private key on the certificate. Once the user’s credentials have been obtained, it is necessary to verify whether the user has previously registered with the system in order to allow him/her access.

• Certificate Management. The state of the certificates is verified using Web Service, by means of a platform that manages signatures that is installed in an internal domain. When transference begins, and if it has been authorised, a chain of characteristics is
generated as a witness or credential, which will be presented with each of the fragments sent. This guarantees that the user’s identity is not supplanted and that the data regarding the session are not manipulated.

- Secure Channel. Communication should be coded throughout the process, thus guaranteeing confidentiality by means of protocols that will ensure that the circuit by which the information will be transmitted (HTTP over SSL, known as HTTPS) is secure.

The objective of this evaluation phase is to attempt to establish a security level for the mechanisms or sets of mechanisms. This will make it possible to identify the level of quality when evaluating the security of the system once it is incorporated and will show the level of verification and assurance covered or satisfied when these mechanisms are incorporated. It is, therefore, possible to use the previously defined metrics to evaluate whether access via certificates has been tested internally or by third parties, or not controlled, and whether these certificates have been validated and are correctly controlled for their use, or whether or not they have been developed by third parties. It is also possible to verify whether or not there are specific tools in charge of certain functions for this mechanism. One level of quality or another can, therefore, be obtained according to the criteria of each of the metrics defined.

The results obtained after applying the metrics to each of the categories of the security mechanisms identified previously are shown in Tables 6-5, 6-6 and 6-7.

<table>
<thead>
<tr>
<th>Secure Login</th>
<th>Quality family</th>
<th>Justification</th>
<th>Quality level</th>
</tr>
</thead>
<tbody>
<tr>
<td>QAM_COV</td>
<td>All key functionalities of the deployed security mechanisms, user whitelist and login with digital certificate, are verified and checked in the evaluation process and cloud controls related to them are identified.</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>QAM_DPT</td>
<td>All known key properties for each one of the key functionalities for the security mechanisms, user whitelist and login with digital certificate, are verified with help of other systems such as Tomcat and PKCS#11.</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>QAM_RIG</td>
<td>A clear verification requirement document exists for the secure login, and an automatic tool for this purpose is used to verify the credentials, certificates and digital signature.</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>QAM_IND</td>
<td>The verification is partly performed by those who set up the security mechanism deployment features, while other properties are assessed by a third-party (external to the organisation or provided internally and who did not take part in the deployment).</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>
### Certificate Management

<table>
<thead>
<tr>
<th>Quality family</th>
<th>Justification</th>
<th>Quality level</th>
</tr>
</thead>
<tbody>
<tr>
<td>QAM_COV</td>
<td>Many key functionalities of the security mechanisms deployed, such as custom certificates, advanced certificate signature frameworks, custom network settings and verification token and digital signatures, are verified and checked in the evaluation process and cloud controls related to them are identified.</td>
<td>2</td>
</tr>
<tr>
<td>QAM_DPT</td>
<td>All known key properties for each one of the key functionalities for these security mechanisms for this category are verified with the help of other systems, such as a platform for signature management deployed in one of the organisation’s internal domains.</td>
<td>4</td>
</tr>
<tr>
<td>QAM_RIG</td>
<td>A clear verification requirement document exists for the certificate management category, and an automatic tool for this purpose is used for the verification of the certificates and signature digital.</td>
<td>3</td>
</tr>
<tr>
<td>QAM_IND</td>
<td>The verification is partly performed by those who set up the security mechanism deployment features, while other properties are assessed by a third-party (external to the organisation or internal, provided they did not take part in the deployment).</td>
<td>2</td>
</tr>
</tbody>
</table>

### Secure Channel

<table>
<thead>
<tr>
<th>Quality family</th>
<th>Justification</th>
<th>Quality level</th>
</tr>
</thead>
<tbody>
<tr>
<td>QAM_COV</td>
<td>All key functionalities of the security mechanism deployed, such as the encrypted channel, are verified and checked during the evaluation process, and cloud controls related to them, are identified.</td>
<td>3</td>
</tr>
<tr>
<td>QAM_DPT</td>
<td>All known key properties for each one of the key functionalities for the security mechanism of the encrypted channel are verified with the help of other protocols, such as HTTPS and SSL.</td>
<td>4</td>
</tr>
</tbody>
</table>
A clear verification requirement document exists for the secure channel, and an automatic tool for this purpose is used to verify the credentials.

The verification is partly performed by those who set up the security mechanism deployment features, while other properties are assessed by a third-party (external to the organisation or internal, provided they did not take part in the deployment).

Bearing in mind the levels of quality and assurance of security, along with the combinations (see Table 4-6) of the values of the metrics for each category, we can therefore conclude that the level of quality as regards verification and assurance for the REM model is Level QL4, as is indicated in Table 6-8.

<table>
<thead>
<tr>
<th>QAM_COV</th>
<th>QAM_DPT</th>
<th>QAM_RIG</th>
<th>QAM_IND</th>
<th>QL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Secure Login</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Certificate Management</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Secure Channel</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>

It is, therefore, possible to conclude that the verification follows a well-defined process with a usage of software tools, which cover most of the relevant parts (functions and properties) of the security mechanism and that the associated cloud controls are identified and deployed in the new cloud environment.

6.2.2.5.2. Task 5.2 Definition of new requirements

This application is currently in production. It has not, as yet, been necessary to incorporate new requirements.
6.3. **WECO**

When Law 39/2015 of the 'Procedimiento Administrativo Común de las Administraciones Públicas' (Common Administrative Procedure for Public Administrations) came into force on 1st October 2016, it supposed a revolution in the way in which the files of the various public administrations were electronically managed. The system that allows the management of files in accordance with Law 39/2015 at CNMC is denominated as WECO.

It is of particular note that the aforementioned legislation encourages the sharing and reuse of applications by diverse administrations. In particular, Law 11/2007 regarding the public’s access to Public Services states in “Article 45. The reuse of systems and applications owned by the Administration (that) those administrations that are titleholders of the intellectual property rights of applications or whose development has been contracted, will be able to make them available to any other Administration without consideration or agreement”.

During the development of the aforementioned Article 45 of Law 11/2007, the ‘Esquema Nacional de Interoperabilidad’ (National Schema for Interoperability - Royal Decree 4/2010) dedicates Chapter VIII to the reuse “of associated applications and documents, and of other objects of information of whose intellectual property rights the public Administrations are titleholders”.

The fact that legislation encourages the reuse of code and that various public administrations do not have the tools required to adequately manage files in accordance with Law 39/2015, therefore, signifies that it has been necessary to migrate WECO to the cloud in order to make it available to all those administrations that need it.

Section 6.3 is structured as follows: First, in Section 6.3.1, we provide a brief description of the system in which SMiLe2Cloud has been applied and the problem to solve, after which, in Section 6.3.2, we present the application of the SMiLe2Cloud activities, along with the application of the SMiLe2Cloud-Ecosystem tool developed to support those activities, which were explained in Chapter 5.

In this case study we have reused processes and decisions of the first case study. For this reason, some details are not included in this second case study.

6.3.1. **Description of the system**

The ‘Gestión de Expedientes de la Comisión Nacional de los Mercados y la Competencia’ (Management of the National Commission for Markets and Competition’s Files – from here on, WECO) application is a Web application that is specialised in the management of electronic files. Its objective is to ‘be’ the file rather than being the reflection of a file written on paper.

In this respect, it is worth highlighting that at present it successfully manages files consisting of tens of thousands of pages, some of which involve more than 100 companies in different roles. It facilitates the management of associated documentation, controls the confidentiality of documents (including company exceptions), controls information requirements and companies that have not responded, controls the acknowledgement of receipt of pending receipts of notifications issued by any medium (post office, courier, fax,
telematic route), controls expiry dates along with the internal deadlines of the records, allows the registration of contact data of all the companies, maintaining the integrity of each record, makes it possible to obtain lists of files that are being processed, have been finalised, are classified by typology, indicators, statistics, graphs and all types of data that are useful to discover the current situation, and can also be adapted to small files owing to the fact that its conceptual base is very simple.

WECO allows the interrelation of records corresponding to different administrative procedures, and this integration of information provides a new work dimension, since relating files with each other allows the situation of the companies involved to be known in greater detail. It favours the study of economic operations of the market and their analysis, and will consequently be in a position to take measures and provide solutions that are in greater accordance with the reality of the market.

WECO has also proved useful in the management of the lifecycles of the administrative procedures that are fundamental to the CNMC.

6.3.1.1. **Functional description**

The general schema of the CNMC applications needed to comply with Law 39/2015 are shown in Figure 6-17.

- **Register**: The CNMC's electronic register is a register for the presentation of documents for transference to not only the CNMC, but to any other administrative organisation of the State General Administration, public Organisation or Entity linked to or depending on them.
- **Electronic office**: sede.cnmc.gob.es provides the point of access for the procedures and services electronically provided by the CNMC.
- **WECO**: this is the application that manages all the CNMC's files.
- **Distributor**: this application makes it possible to electronically distribute all the documents that enter via the Register or Electronic Headquarters. It is part of WECO.
- **Web**: all the records/files that form part of WECO are automatically published at: www.cnmc.es.
- **Notifications Portal**: the portal in which it is possible to consult various types of notifications placed at the disposal of CNMC users is: cnmc.gob.es.
6.3.1.1.1. **Focused on files**

Each file contains the complete history of a case. The file could, for example, manage a grant, a record regarding sanctions, a parliamentary question, etc.

Each file stores the most important milestones (events) that have occurred. It may, for example, store the application for the grant, whether it has been conceded or refused, etc.

Each file belongs to a sphere (a sphere can be identified as an administrative procedure). Examples of spheres are, for example, grants, reports, sanctions, resources, etc.

Each sphere is subdivided into types of file. The grant sphere is, for example, divided into university grants, secondary school grants, etc.

6.3.1.1.2. **No workflows**

WECO is a totally flexible application as regards the order in which information can be introduced. Any piece of data can be introduced at any time at any point of any file.
This flexibility signifies that the application is more resistant to organisational change or moments of crisis, situations that will, sooner or later, always occur.

The non-definition of workflows policy is accompanied by a collaborative approach, signifying that each person’s work is watched over (in the good sense of the word) by many other people.

6.3.1.1.3. **Strictly Web application**

The use of the application requires only the availability of a Web browser that can be connected to the server in which the application is installed. Special care was taken during its development to ensure that it works with all browsers (Internet explorer, Firefox, Chrome), without the need to use any other additional accessory, plugin or installation.

The operative system from which the Web browser is used is not important either, and can be Windows, Mac, Linux, Android, etc. If the application is installed in the interior of an organisation, and is inaccessible via the Internet, any type of generic Virtual Private Network (VPN) that habilitates the connection between an Internet user’s PC, tablet or laptop and the server in which WECO is installed will permit the use of the application via the browser.

6.3.1.1.4. **With security in mind**

Security is one of the points to have been borne in mind from the outset, thus leading to the establishment of a flexible integrated security system that is integrated with the organisation’s directory.

The security of WECO is based on groups from the Microsoft Active Directory, and having this directory is, therefore, an important requirement.

There are 3 levels of permits, which are different for each sphere:

- **Read permits**: these allow the files to be read, without the possibility of making any modifications.
- **Write permits**: these allow access to the files and permit all the insertions to and modifications and eliminations of any piece of data to be carried out.
- **Administration permits**: these allow the files to be modified, in addition to permitting the modification of that particular sphere.

It is important to stress that the application has a powerful restriction engine, which allows restrictions concerning existing spheres, files and events to be defined. A restriction is a small group of conditions. If all of these are fulfilled, a restriction is applied, signifying that any restricted content is hidden from those people who have been defined.

There are many types of restriction of a diverse nature, and those shown as follows are, therefore, only a few examples. These are restrictions that can define:

- That a group of people does not have access to a certain type of files in a sphere.
- That a group of people does not have access to certain files that are in a certain phase.
• That a group of people can see certain files, but only after a particular event.
• That a group of people can only see certain files if something has occurred in them.
• That each person only has access to the files that s/he has defined, as the person responsible for that file, or its unit.

6.3.1.2. Architecture

Details of the most important characteristics of the architecture of the application are shown below:

• Operating system: Windows Server 2016
• Web development:
  • Server language: ASP.Net 4.5 (Visual Basic language).
  • Client language: HTML5 + JavaScript + CSS. The jQuery, Knockout and Vue libraries are used.
• Database: Microsoft SQL Server 2016.
• Web server: Microsoft Internet Information Services (IIS 10).
• Information repository: FTP Server.
• Authentication and Authorisation: Integrated with Microsoft Active Directory.

The current infrastructure in the CNMC, in which the application is deployed, is shown in Figure 6-18.

![Figure 6-18. Current WECO architecture](image-url)
6.3.1.3. **Interface**

The interface of the main screen of WECO is shown in Figure 6-19. It consists of:

- The various spheres (administrative proceedings), which are on the left.
- A searcher to locate files.
- A series of panels (control panels) that are highly useful for the Directive personnel.
- A list of the files most recently visited.

![Figure 6-19. WECO Interface](image)

6.3.2. **Application of SMiLe2Cloud**

This section shows the application of SMiLeCloud in order to migrate the WECO system to the cloud. Each of the activities will be analysed in turn, specifying their principal input and output flows.

6.3.2.1. **Analysis Activity**

Since the application was built using .NET technology, it is not possible to apply the MARBLE and IBOPROFEN tools, since they currently support only java.
For this reason, in this case study it was first necessary to obtain the functional description of the system expressed as a goal model, since the extraction activity could not be carried out, and the reverse engineering is, therefore, ignored in this migration. This task was carried out together with the analysis of the security requirements.

6.3.2.1.1. Task 2.1: Analysis of Security Requirements

As explained above, the first step consisted of defining the goal model. Since WECO is a tremendously complex system, we carried out a simplified study of some of the principal functionalities that relate the user to the exterior: allow user login, perform digital signature and upload files.

![WECO SMiLe2Cloud goal model](image)

The following goals have been identified: file management, allow user login, perform digital signature and upload files.

The result of the application of the analysis of security requirements task to the case study is shown in Figure 6-21. Two threats have been identified (eavesdropping and user impersonation). Moreover, three security constraints have been found: access only to registered users, integrity of files must be protected and confidentiality of file contents.

In addition to this, the following security mechanisms have been identified: alternate certificate stores, active directory, login with digital certificate, advanced certificate signature frameworks, encrypted channel, verification token and digital signature.
6.3.2.1.2. Task 2.2: Align Security Requirements with Cloud Security Controls

The process explained in Section “4.3.2.2.3. Align security mechanism” was carried out for the various security mechanisms: alternate certificate stores, active directory, login with digital certificate, advanced certificate signature frameworks, encrypted channel, verification token and digital signature.

The list of controls that are applicable to these security mechanisms are shown in the table 6-9.
<table>
<thead>
<tr>
<th>Domain</th>
<th>Control</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application &amp; Interface Security</td>
<td>Data Integrity</td>
<td>AIS-03</td>
</tr>
<tr>
<td></td>
<td>Data Security/Integrity</td>
<td>AIS-04</td>
</tr>
<tr>
<td>Audit Assurance &amp; Compliance</td>
<td>Information System Regulatory Mapping</td>
<td>AAC-03</td>
</tr>
<tr>
<td>Business Continuity Management &amp; Operational Resilience</td>
<td>Equipment Location</td>
<td>BCR-06</td>
</tr>
<tr>
<td></td>
<td>Equipment Power Failures</td>
<td>BCR-08</td>
</tr>
<tr>
<td>Data Security &amp; Information Lifecycle Management</td>
<td>Controlled Access Points</td>
<td>DCS-02</td>
</tr>
<tr>
<td></td>
<td>Identification</td>
<td>DCS-03</td>
</tr>
<tr>
<td></td>
<td>Unauthorized Persons Entry</td>
<td>DCS-08</td>
</tr>
<tr>
<td>Datacenter Security</td>
<td>Entitlement</td>
<td>EKM-01</td>
</tr>
<tr>
<td></td>
<td>Key Generation</td>
<td>EKM-02</td>
</tr>
<tr>
<td></td>
<td>Storage and Access</td>
<td>EKM-04</td>
</tr>
<tr>
<td>Security Identity &amp; Access Management</td>
<td>Credential Lifecycle / Provision Management</td>
<td>IAM-02</td>
</tr>
<tr>
<td></td>
<td>Diagnostic / Configuration Ports Access</td>
<td>IAM-03</td>
</tr>
<tr>
<td></td>
<td>Segregation of Duties</td>
<td>IAM-05</td>
</tr>
<tr>
<td></td>
<td>User Access Authorization</td>
<td>IAM-09</td>
</tr>
<tr>
<td></td>
<td>User Access Revocation</td>
<td>IAM-11</td>
</tr>
<tr>
<td></td>
<td>User ID Credentials</td>
<td>IAM-12</td>
</tr>
<tr>
<td></td>
<td>Utility Programs Access</td>
<td>IAM-13</td>
</tr>
<tr>
<td>Infrastructure &amp; Virtualization</td>
<td>Vulnerability Management</td>
<td>IVS-05</td>
</tr>
<tr>
<td></td>
<td>Network Security</td>
<td>IVS-06</td>
</tr>
<tr>
<td></td>
<td>OS Hardening and Base Controls</td>
<td>IVS-07</td>
</tr>
<tr>
<td></td>
<td>Production / Non-Production Environments</td>
<td>IVS-08</td>
</tr>
<tr>
<td></td>
<td>VM Security - Data Protection</td>
<td>IVS-10</td>
</tr>
<tr>
<td></td>
<td>Hypervisor Hardening</td>
<td>IVS-11</td>
</tr>
<tr>
<td></td>
<td>Wireless Security</td>
<td>IVS-12</td>
</tr>
<tr>
<td></td>
<td>Network Architecture</td>
<td>IVS-13</td>
</tr>
<tr>
<td>Interoperability &amp; Portability Virtualization</td>
<td>Virtualization</td>
<td>IPY-05</td>
</tr>
<tr>
<td>Mobile Security</td>
<td>Cloud Based Services</td>
<td>MOS-06</td>
</tr>
<tr>
<td></td>
<td>Device Management</td>
<td>MOS-10</td>
</tr>
<tr>
<td></td>
<td>Encryption</td>
<td>MOS-11</td>
</tr>
<tr>
<td></td>
<td>Jailbreaking and Rooting</td>
<td>MOS-12</td>
</tr>
</tbody>
</table>
6.3.2.2. **Design Activity**

The design activity is focused on selecting the service model and the deployment model.

6.3.2.2.1. **Task 3.1 Identification of implementation model and service model**

**Identification of implementation model**

As explained previously, the objective of this migration is possible reuse in other organisations that are dependent on the AGE (General State Administration). As in the previous case, since there is currently no private cloud model in the AGE, the only implementation model that can be applied in our case study is the public cloud.

**Identification of service model**

When choosing the service model, it is important to know that WECO is not currently prepared to be a multi-organisation. If this were the case, then WECO could be provided in SAAS format. But since this is not so, the only way in which to provide the various AGE organizations with WECO is by using the Cloud Infrastructure as a Service (IaaS) model.

6.3.2.2.2. **Task 3.2 Selection of Cloud provider**

**Selection of cloud provider**

As in the previous case, the case study is based solely on two principal cloud provider: Azure and Amazon AWS, since these providers comply with the controls defined in the analysis activity by 100%.

As will be noted in the next table, WECO employs the Microsoft infrastructure.
Table 6-10. Physical infrastructure of WECO

<table>
<thead>
<tr>
<th>Infrastructure</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft IIS application servers</td>
<td>1</td>
</tr>
<tr>
<td>Active directory server</td>
<td>1</td>
</tr>
<tr>
<td>SQL Server database servers</td>
<td>1</td>
</tr>
<tr>
<td>Storage</td>
<td>1 TB</td>
</tr>
<tr>
<td>Backup</td>
<td>1 TB</td>
</tr>
<tr>
<td>VPN</td>
<td>3</td>
</tr>
</tbody>
</table>

A comparison of costs as regards Microsoft Azure and Amazon AWS is shown in the following table.

Table 6-11. Comparison of costs

<table>
<thead>
<tr>
<th>Infrastructure</th>
<th>Azure</th>
<th>AWS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application servers</td>
<td>38,90 €</td>
<td>57,78 €</td>
</tr>
<tr>
<td>Database Servers</td>
<td>63,24 €</td>
<td>231,43 €</td>
</tr>
<tr>
<td>Active Directory Servers</td>
<td>25,72 €</td>
<td>292,8 €</td>
</tr>
<tr>
<td>Storage</td>
<td>91,87 €</td>
<td>107,6 €</td>
</tr>
<tr>
<td>Backup</td>
<td>201,07 €</td>
<td>40,9 €</td>
</tr>
<tr>
<td>VPN</td>
<td>152,74 €</td>
<td>96,9 €</td>
</tr>
<tr>
<td>Total</td>
<td>573,54 €</td>
<td>827,41 €</td>
</tr>
</tbody>
</table>

This study of costs can be complimented with the use of the SWOT analysis of both cloud providers described in Section 6.2.2.3.2.

The Azure cloud offers much more advantageous prices for our case study. This is principally owing to the fact that the WECO application was developed using Microsoft technologies. The Microsoft cloud thus offers much more competitive prices with regard to its own products.

Also, the Azure cloud has the highest level in the National Security Schema of the Spanish National Cryptology Centre. This is very important for the Spanish Public Administration because this is part of the regulatory framework.

For these reasons, therefore, appear logical to choose Microsoft Azure as the provider.
6.3.2.3. Deployment Activity

The deployment activity is focused on developing the deployment specification and the implementation of the system.

6.3.2.3.1. Task 4.1 Deployment Specification

As indicated in Task 4.1 “Deployment Specification”, the Consensus Assessments Initiative Questionnaire (CAIQ) can be used as a basis to identify the security services of the cloud provider chosen, in this case (CAIQ, 2017).

For example, according to the indications in the questionnaire used to guarantee the control “IAM-12 User ID Credentials”, it is necessary to use the Azure Active Directory service.

The same process employed to obtained the list of controls identified was also used to obtain the services that should be installed in Azure:

- Azure Security - General
  - Azure Disk Encryption
  - Log Analytics
- Azure Storage Security
  - Azure Storage Service Encryption
- Azure Database Security
  - Azure SQL Firewall
  - Azure SQL Connection Encryption
  - Azure SQL Authentication
  - Azure SQL Always Encryption
  - Azure SQL Database Auditing
- Azure Identity and Access Management
  - Azure Role Based Access Control
  - Azure Active Directory
- Backup and Disaster Recovery
  - Azure Backup
- Azure Networking
  - Network Security Groups
  - Azure VPN Gateway
6.3.2.3.2. **Task 4.2. Implementation**

This case study was implemented on Microsoft cloud services.

**Functional description**

The technologies used in the original development have allowed its migration to be carried out without the need to make changes to its source code.

The application is, therefore, functionally similar.

**Architecture**

The infrastructure required can, however, be simplified. The minimum infrastructure required to deploy WECO is shown in the Figure 6-22:

![Figure 6-22. Minimum WECO architecture](image)

The minimal WECO architecture is composed by:
- IIS + FTP server
- Database: it was necessary to make some adjustments to the database in order to make use of the Amazon database in service mode.
- Domain controller
- Networks: 3 independent VLANs
The deployment of the WECO infrastructure in Azure is shown in Figure 6-23. The deployment of the infrastructure has been automated. This was done by developing a Visual Studio solution that contains a JSON file that has all the configurations required for the deployment (servers, database, nets, etc.). Deploying the WECO application in Azure was, therefore, as simple as executing the script.

Part of the JSON related to the deployment of WECO, and specifically that regarding the creation of a server in Azure, is shown in Figure 6-25.
The result of installing WECO in Azure is shown in the Figure 6-25:

![Figure 6-25. WECO in Azure](image)

**Interface**

The result of installing WECO in Azure is shown in the Figure 6-25:

**Service Level Agreements**

Finally, it is necessary to sign service level agreements with the provider. In this case, the service level agreement signed contains:

- Supports the first level, which includes priority attention and telephonic support.
6.3.2.4. **Evaluation Activity**

This activity consisted of carrying out the same process as that realised in the first case study for the REM system (see Section 6.2.2.5). This meant that we could reuse the evaluation carried out since we were dealing with similar mechanisms, with the exception of certain small changes, such as the incorporation of the “alternate certificate stores” and “active directory” mechanisms and the elimination of others. The set of mechanisms obtained for WECO by following the same structure as in the first case study regarding REM is, therefore, shown in Table 6-12.

<table>
<thead>
<tr>
<th>Category</th>
<th>Security Mechanism</th>
</tr>
</thead>
<tbody>
<tr>
<td>Secure Login</td>
<td>user whitelist</td>
</tr>
<tr>
<td></td>
<td>login with digital certificate</td>
</tr>
<tr>
<td>Certificate Management</td>
<td>custom certificates</td>
</tr>
<tr>
<td></td>
<td>advanced certificate signature frameworks</td>
</tr>
<tr>
<td></td>
<td>custom network settings</td>
</tr>
<tr>
<td></td>
<td>verification token and digital signature</td>
</tr>
<tr>
<td>Secure Channel</td>
<td>encrypted channel</td>
</tr>
</tbody>
</table>

Upon carrying out the evaluation of these mechanisms in the same way as that used for REM, while bearing in mind the characteristics and requirements of WECO, we obtained the values for the metrics shown in Table 6-13. The level of the quality of the WECO security mechanisms is 4.

<table>
<thead>
<tr>
<th></th>
<th>QAM_COV</th>
<th>QAM_DPT</th>
<th>QAM_RIG</th>
<th>QAM_IND</th>
<th>QL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Secure Login</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>Certificate Management</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>Secure Channel</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>
The difference between the values obtained for WECO and those obtained for REM is owing to the new mechanisms incorporated.

- Active directory, which is an external mechanism that depends on the operative system and is verified and tested in all the specific functionalities and properties. This results in a maximum value in the QAM_COV = 3, QAM_DPT = 4 and QAM_RIG = 3 metrics for the Secure Login category.

- Alternate Certificate stores is a mechanism that is internal to the system and which should comply with the same original criteria and restrictions, signifying that it is also validated and verified using the internal tools and procedures and complies with the established standards, and it can thus be considered that all its functionalities and properties comply at the maximum level, as is shown by the values in QAM_COV = 3, QAM_DPT = 4 and QAM_RIG = 3.

The value of QAM_IND is 2 owing to the fact that there is not total independence between those people implied in the development of the mechanism and those that audit and verify its functionalities and properties, who are generally people from the organisation itself, although they are not involved in the project. This value is the metric that prevents the System and the level of assurance and verification of the security of its mechanisms from attaining the maximum value of 5 according to the criteria shown in Table 4-6 in Section 4.3.5.1 of Chapter 4.

Therefore, we can conclude that the verification follows a well-defined process with a usage of software tools, which cover most of the relevant parts (functions and properties) of the security mechanism and the cloud controls associated are identified and deployed in the new cloud environment.

### 6.4. Lessons Learned

The development of the empirical evaluation presented in this chapter has allowed us to put the SMiLe2Cloud framework into practice in two real use environments. This work has enabled us to achieve two objectives. On the one hand, to validate both the integral functioning of the framework and the suitability of the activities and tasks proposed; and on the other, to achieve the secure migration of two systems that required modernisation to cloud computing.

SMiLe2Cloud proposes a long-term process whose successive iterations guarantee that an appropriate degree of security will be achieved for the migration of a legacy application to cloud computing. Nevertheless, the empirical evaluation described in both cases represents the first iteration in the process. Despite the fact that these cases did not allow us to reach long-term conclusions, they showed us a sufficient amount for us to attain representative results regarding the research questions proposed.

Of the lessons learned, we should highlight the importance of introducing additional metrics in all its phases and components, and not only for its security mechanisms.

The main drawback found by the participants in the empirical evaluation process was the lack of specific proposals as regards support tools. The sphere of SMiLe2Cloud is so broad and integral that it is not practical to propose the development of an automatic tool that can support the entire framework.
Throughout the empirical evaluation we also identified that the personnel on the project have to make much more effort during the first three activities. The tasks in the activities related to extraction, analysis and design required more work. This behaviour could be attributed to the innovative nature of this type of services in the organisation in question, along with the precautions taken in order to guarantee security.
7. Conclusions
This chapter shows the conclusions obtained after carrying out the work presented in this doctoral thesis.

These conclusions are structured as follows: We first provide an analysis of the attainment of the objectives set out at the beginning of the thesis, after which we describe the principal contributions made by this doctoral thesis. We then compare the results obtained and indicate those papers regarding this subject that have been published. Finally, we propose future lines of work that still remain open after carrying out this research.

7.1. Analysis of Objectives Attained

This doctoral thesis proposes the SMiLe2Cloud framework, whose purpose is to migrate legacy security systems to the cloud.

The origin of this research work is the principal motivation behind this thesis, as stated in Chapter 1. This objective is depicted in Figure 7-1.

![The definition of a methodological framework to support migration of legacy systems to secure cloud computing](Figure 7-1. Global objective of the thesis)

In order to achieve this global objective, a set of partial objectives was proposed in Section 1.5 whose aim was to collaborate together in the development of a process that would satisfy the principal objective proposed. These partial objectives are reviewed as follows, and their attainment is analysed.

**Objective 1.** Analyse the existing proposals related to the frameworks, processes and methodologies that allow the security migration of legacy systems to cloud computing.

Chapter 3 shows the results of an analysis of the State of the Art carried out, which comprises the most relevant proposals regarding the area of this thesis that currently exist. This analysis made it possible to discover a high number of guidelines, standards and security processes that could be used to migrate legacy security systems to cloud computing.

The systematic literature review carried out enabled us to observe that the proposals in this sphere are focused on either the processes employed to migrate legacy systems to cloud computing or the security of cloud computing environments. It was not, however, possible to locate any works that unite both spheres from a global and valid perspective.

This analysis, therefore, allowed us to discover an area of research on which the efforts of this thesis could be focused, in addition to facilitating the identification of the principal contributions of the existing proposals, such that the work carried out can be aligned with them and take advantage of their benefits. This objective is, therefore, considered to have been fully covered.
Objective 2. Study the different techniques and tools that support reengineering, specifically those that support software modernisation and business process archaeology.

Section 3.3 shows the results obtained after studying the different techniques and tools that support reengineering, and particularly those that support the modernisation of software and business process archaeology.

The systematic literature review led us to observe that some of the existing proposals could be used in the extraction activity of the SMiLe2Cloud framework.

We therefore consider that this objective has been completely covered.

Objective 3. Study the specific security proposals used to analyse security requirements and needs from the first stages of the software lifecycle.

Section 3.2 shows the results obtained after studying the specific security proposals in order to analyse the security requirements and needs from the first stages of the software lifecycle.

The systematic literature review enabled us to observe that some of the existing proposals could be used in the extraction activity of the SMiLe2Cloud framework.

This objective is, therefore, considered to have been fully covered.

Objective 4. Define the activities, tasks, steps, actors and artefacts needed to design an engineering process for the security migration of legacy systems to cloud computing.

Chapter 4 presents the SMiLe2Cloud framework, whose purpose is to ensure security migration from legacy systems to cloud computing.

The SMiLe2Cloud framework has been formally modelled following the SPEM 2.0 (Software & Systems Process Engineering Meta-Model), developed by the OMG (Object Management Group) (OMG, 2008). The use of this modelling allowed us to provide a homogeneous and standardised representation of the processes, which can be used and managed by means of automated electronic repositories, and additionally facilitates the reuse of contents by employing tools external to the framework.

This modelling was principally focused on two spheres: the structure of the activities and tasks proposed, and the repository of artefacts, which contains the elements required to develop the process described.

With regard to modelling the activities and tasks, we used the standardised SPEM notation to define its content in a homogeneous manner. According to this notation, the formal specification of each task includes the roles of the personnel that will take part in it, the artefacts involved as regards both inputs to and outputs from the process, the steps proposed for the execution of the task, and the guidelines, techniques or improvements suggested in order to support its development.

The SMiLe2Cloud framework totally fulfils this objective, since it has processes whose value for this purpose has been proved, and which are widely used by the community.
Objective 5. Ensure the alignment of the process with international security standards and ensure its compliance with them.

The principal international security standards were taken into consideration when designing the SMiLe2Cloud framework. The most relevant references, as is detailed in Chapter 4, are the following:

- ISO/IEC 27001 (ISO/IEC 27001, 2013), which specifies the requirements that an organisation should fulfil in order to implement and operate the ISMS (ISMS — Information Security Management System).
- ISO/IEC 27002 (ISO/IEC 27002, 2013), which describes numerous security measures, such that they can be used to manage a security system in an organisation.
- ISO/IEC 27017 (ISO/IEC 27017, 2015), which provides guidelines for information security controls applicable to the provision and use of cloud services by providing additional implementation guidance for relevant controls specified in ISO/IEC 27002; and additional controls with implementation guidance that specifically relate to cloud services.
- ISO/IEC 27018 (ISO/IEC 27018, 2014), which establishes commonly accepted control objectives, controls and guidelines with which to implement measures intended to protect Personally Identifiable Information (PII) for the public cloud computing environment.
- ISO/IEC 27036 (ISO/IEC 27036, 2014), which is focused on the Information Security of the relationships with the providers.
- Cloud Security Alliance Security Guidelines (Cloud Security Alliance, 2011), which concern questions specifically related to security in the cloud.

The integration of SMiLe2Cloud into these references guarantees that the proposed framework is aligned with widely disseminated and accepted processes, whilst additionally facilitating its understanding and use by organisations and professionals who are already familiar with these standards and guidelines.

These results show that the SMiLe2Cloud framework satisfies this objective as regards both its integration with current security standards and the possibility of adapting it in the future in order to enable its evolution.

Objective 6. Design and implement a tool prototype in order to provide the proposal with support.

Since the SMiLe2Cloud framework was defined on the basis of standardised models and languages, it is possible to employ certain tools that allow the semi-automatic management of particular tasks in the process.

In Chapter 6, an analysis was carried out of the tools that are available in order to provide organisations with those that may have most value as regards executing SMiLe2Cloud. It is reasonable to take advantage of the tools that are currently available for certain aspects of the process. The integrated and coordinated use of these tools constitutes a functional alternative by which to attain adequate support for the methodology proposed.

The principal contribution of the solution presented is the integration of these tools, which together enable the security migration process defined in the SMiLe2Cloud framework to be guided. Each tool in itself can be used in only one of the process activities. However, the
solution proposed permits the use of the SMiLe2Cloud framework in all of its activities, from the extraction activity to the evaluation activity.

The majority of this objective as regards supporting the development of the SMiLe2Cloud framework is covered thanks to this compendium of tools.

Objective 7. Validate the proposal through its practical application in real scenarios.

The practical validation of the SMiLe2Cloud framework was carried out using an empirical evaluation process. This empirical evaluation process, which is described in Chapter 5, was developed on two systems that are currently in production at the CNMC (Comisión Nacional de los Mercados y la Competencia – National Commission for Markets and Competition).

The results of this process enabled us to provide satisfactory responses to the research questions and, therefore, demonstrate the practical validity of the SMiLe2Cloud framework.

It is possible to conclude that SMiLe2Cloud totally fulfils its objective and can, in practice, be implemented to obtain the desired results.

In conclusion, bearing in mind that we have been able to satisfy the various partial objectives proposed, it is possible to state that the principal objective, which was specified as: “Define a methodological framework with which to enable the security migration of legacy systems to cloud computing, defining the activities, tasks and steps to be performed, the elements used in each task, the artefacts used in each activity and the people responsible involved” has also been fulfilled.

As a result, it is also possible to state that the initial hypothesis of this doctoral thesis, which was defined as “The migration of legacy systems to secure cloud computing can be benefited by methodological frameworks” is affirmative, as has been demonstrated.

7.2. Principal Contributions

The principal contributions made by this doctoral thesis are listed as follows:

1. The realisation of a State of the Art of processes related to the migration of legacy Security Systems to cloud computing. This analysis was developed in accordance with the protocol for Systematic Literature Reviews, and its results were presented in Chapter 3 of this thesis.

2. The proposal of the SMiLe2Cloud framework as an instrument that will provide organizations with a process with which to migrate legacy Security Systems to cloud computing, and which is described in Chapter 4. This proposal integrates the most relevant aspects of the security standards and best practices in this area, thus giving it greater consistency and facilitating its implementation.

SMiLe2Cloud has the following activities, which cover the different stages of the lifecycle of these services:

   - Activity 1: Extraction
   - Activity 2: Analysis
• Activity 3: Design
• Activity 4: Implementation
• Activity 5: Evaluation

In order to facilitate the implementation and reuse of SMiLe2Cloud, all of its elements have been modelled in conformance with the SPEM 2.0 specification.

3. Chapter 5 provides an analysis of the set of tools that can be used to set up the SMiLe2Cloud framework, with the purpose of facilitating the execution of the proposed processes.

4. The realisation of an empirical evaluation of the proposal, which is presented in Chapter 6, during which the viability of its use in practice was validated, and for which favourable results were obtained.

7.3. Contrast of Results

The various contributions and proposals included in this thesis have been presented in several different scientific forums. The acceptation of each part of the research by the scientific community provides the work developed with additional quality. A summary of the publications achieved within the framework of this doctoral thesis, classified by type of publication, is shown in Table 7-1.

<table>
<thead>
<tr>
<th>Type of Publication</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>International Conferences</td>
<td>6</td>
</tr>
<tr>
<td>National Conferences</td>
<td>1</td>
</tr>
<tr>
<td>Papers in International Journals (JCR)</td>
<td>1</td>
</tr>
<tr>
<td>Papers in National Journals</td>
<td>1</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td><strong>9</strong></td>
</tr>
</tbody>
</table>

Each of these publications was focused on a different aspect of the contributions made by this thesis. A classification of these publications according to the different contributions made is shown in Table 7-2. Each publication has been identified with a code, consisting of the year and the initials of the publication and which will later be used when describing them in detail.

<table>
<thead>
<tr>
<th>Contribution</th>
<th>Publications</th>
</tr>
</thead>
<tbody>
<tr>
<td>State of the Art</td>
<td>WOSIS 2014</td>
</tr>
<tr>
<td>SMiLe2Cloud framework</td>
<td>RECSI 2014</td>
</tr>
<tr>
<td></td>
<td>INCIBE 2014</td>
</tr>
<tr>
<td></td>
<td>CAISE - WISSE 2015</td>
</tr>
</tbody>
</table>
### 7.3.1. International Conferences and Workshops

Table 7-3. Publications in international conferences

<table>
<thead>
<tr>
<th>Code</th>
<th>Publication</th>
</tr>
</thead>
</table>
7.3 Contrast of Results

7.3.1. National Conferences and Workshops

Table 7-4. Publications in international conferences

<table>
<thead>
<tr>
<th>Code</th>
<th>Publication</th>
</tr>
</thead>
</table>

7.3.2. International Journals

Table 7-5. Publications in international journals

<table>
<thead>
<tr>
<th>Code</th>
<th>Publication</th>
</tr>
</thead>
</table>

7.3.1. National Journals

Table 7-6. Publications in national journals

<table>
<thead>
<tr>
<th>Code</th>
<th>Publication</th>
</tr>
</thead>
<tbody>
<tr>
<td>INCIBE 2014</td>
<td>Luis Marquez Alcañiz, David G. Rosado, Daniel Mellado, Eduardo Fernández-Medina, Hacia una migración de la seguridad de sistemas heredados a la nube, Revista Ciber Seguridad.</td>
</tr>
</tbody>
</table>

7.3.2. Cites

The collaboration with some research groups, especially with the SenSe group of the University of Brighton, has allowed our proposal to be useful for the scientific community.
7.4. **Future Lines of Work**

In addition to satisfying the initial objectives established for this doctoral thesis, the research work developed has also opened up other lines of work on which we can continue working.

The principal areas of knowledge in which additional research can be carried out are related to the complexity and range of the sphere of the SMiLe2Cloud framework, and are principally focused on the activities and tasks in the process, their practical implementation and the tools used.

The main aspects that may constitute lines of future work are shown as follows:

a) **SMiLe2Cloud Model**:

1. Provide details on the execution steps in each task, with an in-depth description of each one, such that organisations can see the framework adoption process in a more simplified manner.
2. Complete and improve the support guidelines for each task, such that they provide more useful information for the organisation using it, including examples and criteria regarding how to act.
3. Define new metrics and additional indicators for the proposals in order to increase the monitoring information for the SMiLe2Cloud framework at both a process and a product level.

4. Identify new tasks that may be necessary for specific cloud computing services according to their service model (IaaS, PaaS, SaaS, etc.) or their deployment model (private, public, hybrid, etc.).

5. Define reusable repositories that will permit the storage of behaviour patterns and configuration registers to enable them to be used in future executions of the process, extracting the experience and decisions made in order to optimise future migrations for similar systems.

b) The practical application of SMiLe2Cloud:

1. Carry out new case studies that will make it possible to validate the utility of the SMiLe2Cloud framework in scenarios other than those detailed in Chapter 6.

c) Support tools for the SMiLe2Cloud framework:

1. Identify existing tools that can be used to support the SMiLe2Cloud framework and can be incorporated into our ecosystem. It might, for example, be possible to identify new tools for the extraction activity that contemplate the analysis of more programming languages, and not just java, in the legacy systems.

2. Develop a new tool that would in itself be capable of encompassing the complete support of the entire SMiLe2Cloud framework at the level of both its repositories and the proposed activities and tasks.
Bibliography
Bibliography


European Commission. (2015). SMART 2013/0043 - Uptake of Cloud in Europe: Communication from the commission to the European Parliament, the council, the European economic and social Committee and the Committee of the regions.


IDC. (2016). *IDC Forecasts Worldwide Cloud IT Infrastructure Market to Grow 24% Year Over Year in 2015*.


ISACA. (2014). *Controls & Assurance in the Cloud: Using COBIT 5*.


Rebollo Martínez, O. (2014). Marco para el Gobierno de la Seguridad de la Información en servicios Cloud Computing. (PhD), University of Castilla-La Mancha.


Statista Inc. (2016). Size of the cloud computing and hosting market market worldwide from 2011 to 2019


VentureBeat. (2016). Amazon Web Services brings in $2.4B in revenue in Q4 2015, up 69% over last year.


Appendixes
Appendix I: List of Acronyms

ADM  Architecture Driven Modernization
API  Application Programming Interface
CCM  Cloud Control Matrix
CCN  Centro Criptológico Nacional
CEO  Chief Executive Officer
CERT Computer Emergency Response Team
CIO  Chief Information Officer
CMM  Capability Maturity Model
CMMI Capability Maturity Model Integration
COBIT Control Objectives for Information and related Technology
CSA  Cloud Security Alliance
CSP  Cloud Service Provider
ENISA European Network and Information Security Agency
FedRAMP Federal Risk and Authorization Management Program
GDP  Gross Domestic Product
IaaS  Infrastructure as a Service
IPSec  Internet Protocol Security
IS  Information Systems
ISACA Information Systems Audit and Control Association
ISF  Information Security Forum
ISGTF Information Security Governance Task Force
ISMS Information Security Management System
LIS  Legacy Information Systems
MAGERIT Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información
NIST National Institute of Standards and Technology
OMG Object Management Group
PaaS Platform as a Service
PMI Project Management Institute
ROI  Return on investment
SaaS  Software as a Service
SE  Software Engineering
SEI  Software Engineering Institute
SE AR Action-Research in Software Engineering
SMile2Cloud Secure Migration of a Legacy information system to the Cloud
SPERM  Software & Systems Process Engineering Meta-Model
STAR CSA Security, Trust & Assurance Registry
UML  Unified Modelling Language
## Appendix II: List of security controls of CCM v3 aligned with RMIAS

<table>
<thead>
<tr>
<th>Category</th>
<th>Subcategory</th>
<th>SaaS</th>
<th>PaaS</th>
<th>IaaS</th>
<th>Security goals dimensions</th>
<th>Security countermeasure dimensions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application &amp; Interface Security</td>
<td>Application Security</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Organisational</td>
</tr>
<tr>
<td>Customer Requirements</td>
<td>Customer Access Requirements</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Legal</td>
</tr>
<tr>
<td>Data Integrity</td>
<td>Data Integrity</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability/Integrity</td>
<td>Technical</td>
</tr>
<tr>
<td>Data Security/Integrity</td>
<td>Data Security/Integrity</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Confidentiality/Integrity</td>
<td>Organisational</td>
</tr>
<tr>
<td>Audit Planning</td>
<td>Audit Planning</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Auditability</td>
<td>Organisational</td>
</tr>
<tr>
<td>Independent Audits</td>
<td>Independent Audits</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Auditability</td>
<td>Organisational</td>
</tr>
<tr>
<td>Information System Regulatory Mapping</td>
<td>Information System Regulatory Mapping</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability/Integrity</td>
<td>Organisational</td>
</tr>
<tr>
<td>Business Continuity Management &amp; Operational</td>
<td>Business Continuity Planning</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Organisational</td>
</tr>
<tr>
<td>Resilience</td>
<td>Business Continuity Testing</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Organisational</td>
</tr>
<tr>
<td>Datacenter Utilities / Environmental Conditions</td>
<td>Datacenter Utilities / Environmental Conditions</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Availability</td>
<td>Organisational</td>
</tr>
<tr>
<td>Documentation</td>
<td>Documentation</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>Organisational</td>
</tr>
<tr>
<td>Environmental Risks</td>
<td>Environmental Risks</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Availability</td>
<td>Organisational</td>
</tr>
<tr>
<td>Equipment Location</td>
<td>Equipment Location</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Availability</td>
<td>Technical</td>
</tr>
<tr>
<td>Equipment Maintenance</td>
<td>Equipment Maintenance</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Availability</td>
<td>Organisational</td>
</tr>
<tr>
<td>Equipment Power Failures</td>
<td>Equipment Power Failures</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Availability</td>
<td>Technical</td>
</tr>
<tr>
<td>Category</td>
<td>Policy ID</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Organisational</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>-----------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>-----------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>Impact Analysis</td>
<td>BCR-09</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Organisational</td>
</tr>
<tr>
<td></td>
<td>BCR-10</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Organisational</td>
</tr>
<tr>
<td></td>
<td>BCR-11</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Availability</td>
<td>Organisational</td>
</tr>
<tr>
<td>Change Control &amp; Configuration Management</td>
<td>CCC-01</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Organisational</td>
</tr>
<tr>
<td></td>
<td>CCC-02</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Organisational</td>
</tr>
<tr>
<td></td>
<td>CCC-03</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Organisational</td>
</tr>
<tr>
<td></td>
<td>CCC-04</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Organisational</td>
</tr>
<tr>
<td></td>
<td>CCC-05</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Organisational</td>
</tr>
<tr>
<td>Data Security &amp; Information Lifecycle Management</td>
<td>DSI-01</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Confidentiality</td>
<td>Legal</td>
</tr>
<tr>
<td></td>
<td>DSI-02</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Organisational</td>
</tr>
<tr>
<td></td>
<td>DSI-03</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Legal</td>
</tr>
<tr>
<td></td>
<td>DSI-04</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Confidentiality</td>
<td>Integrity</td>
</tr>
<tr>
<td></td>
<td>DSI-05</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Confidentiality</td>
<td>Legal</td>
</tr>
<tr>
<td></td>
<td>DSI-06</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Privacy</td>
<td>Legal</td>
</tr>
<tr>
<td></td>
<td>DSI-07</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Confidentiality</td>
</tr>
<tr>
<td>Datacenter Security</td>
<td>DCS-01</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Confidentiality</td>
<td>Legal</td>
</tr>
<tr>
<td></td>
<td>DCS-02</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Organisational</td>
</tr>
<tr>
<td></td>
<td>DCS-03</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Technical</td>
</tr>
<tr>
<td>Appendix II</td>
<td>217</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Off-Site Authorization</strong></td>
<td>DCS-04</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Organisational</td>
<td></td>
</tr>
<tr>
<td><strong>Off-Site Equipment</strong></td>
<td>DCS-05</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Organisational</td>
<td></td>
</tr>
<tr>
<td><strong>Policy</strong></td>
<td>DCS-06</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Organisational</td>
</tr>
<tr>
<td><strong>Secure Area Authorization</strong></td>
<td>DCS-07</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Organisational</td>
</tr>
<tr>
<td><strong>Unauthorized Persons Entry</strong></td>
<td>DCS-08</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Technical</td>
</tr>
<tr>
<td><strong>User Access</strong></td>
<td>DCS-09</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Organisational</td>
</tr>
</tbody>
</table>
| **Encryption & Key Management** | Entitlement | EKM-01 | X | X | X | Confidentiality
Integrity
Non-repudiation
Authenticity/Trustworthiness | Technical |
| **Key Generation** | EKM-02 | X | X | X | Confidentiality
Integrity
Non-repudiation
Authenticity/Trustworthiness | Technical |
| **Sensitive Data Protection** | EKM-03 | X | X | X | Confidentiality
Integrity
Non-repudiation
Authenticity/Trustworthiness Auditable | Organisational
Legal |
| **Storage and Access** | EKM-04 | X | X | X | Confidentiality
Integrity
Non-repudiation
Authenticity/Trustworthiness | Technical |
| **Governance & Risk Management** | Baseline Requirements | GRM-01 | X | X | X | Confidentiality
Integrity
Authenticity/Trustworthiness | Organisational
Legal |
| **Data Focus Risk Assessments** | GRM-02 | X | X | X | Confidentiality
Integrity
Non-repudiation
Authenticity/Trustworthiness Auditable | Organisational
Legal |
<p>| <strong>Management Oversight</strong> | GRM-03 | X | X | X | Accountability | Human-oriented |
| <strong>Management Program</strong> | GRM-04 | X | X | X | Authenticity/Trustworthiness | Organisational |</p>
<table>
<thead>
<tr>
<th>Management Support/Involvement</th>
<th>GRM-05</th>
<th>X</th>
<th>X</th>
<th>X</th>
<th>Accountability</th>
<th>Organisational</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy</td>
<td>GRM-06</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Privacy</td>
<td>Organisational</td>
</tr>
<tr>
<td>Policy Enforcement</td>
<td>GRM-07</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Privacy</td>
<td>Organisational</td>
</tr>
<tr>
<td>Policy Impact on Risk Assessments</td>
<td>GRM-08</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Organisational</td>
</tr>
<tr>
<td>Policy Reviews</td>
<td>GRM-09</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>Organisational</td>
</tr>
<tr>
<td>Risk Assessments</td>
<td>GRM-10</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>Organisational</td>
</tr>
<tr>
<td>Risk Management Framework</td>
<td>GRM-11</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>Organisational</td>
</tr>
<tr>
<td>Human Resources</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Asset Returns</td>
<td>HRS-01</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>Human-oriented</td>
</tr>
<tr>
<td>Background Screening</td>
<td>HRS-02</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>Legal</td>
</tr>
<tr>
<td>Employment Agreements</td>
<td>HRS-03</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Confidentiality</td>
<td>Organisational</td>
</tr>
<tr>
<td>Employment Termination</td>
<td>HRS-04</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>Human-oriented</td>
</tr>
<tr>
<td>Mobile Device Management</td>
<td>HRS-05</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Confidentiality</td>
<td>Organisational</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Integrity</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Authenticity/Trustworthiness</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Privacy</td>
<td></td>
</tr>
<tr>
<td>Non-Disclosure Agreements</td>
<td>HRS-06</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Confidentiality</td>
<td>Organisational</td>
</tr>
<tr>
<td>Roles / Responsibilities</td>
<td>HRS-07</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Organisational</td>
</tr>
<tr>
<td>Technology Acceptable Use</td>
<td>HRS-08</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Organisational</td>
</tr>
<tr>
<td>Training / Awareness</td>
<td>HRS-09</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>Legal</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Human-oriented</td>
<td></td>
</tr>
<tr>
<td>User Responsibility</td>
<td>HRS-10</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Organisational</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Privacy</td>
<td>Legal</td>
</tr>
<tr>
<td>Workspace</td>
<td>HRS-11</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Privacy</td>
<td>Organisational</td>
</tr>
<tr>
<td>Security Identity &amp; Audit Tools Access</td>
<td>IAM-01</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Auditability</td>
<td>Organisational</td>
</tr>
<tr>
<td>Access Management</td>
<td>IAM-02</td>
<td>IAM-03</td>
<td>IAM-04</td>
<td>IAM-05</td>
<td>IAM-06</td>
<td>IAM-07</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>Lifecycle / Provision Management</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Organisational</td>
<td>Technical</td>
</tr>
<tr>
<td>Diagnostic / Configuration Ports Access</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Technical</td>
<td></td>
</tr>
<tr>
<td>Policies and Procedures</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Organisational</td>
<td></td>
</tr>
<tr>
<td>Segregation of Duties</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Technical</td>
<td></td>
</tr>
<tr>
<td>Source Code Access Restriction</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Legal</td>
<td></td>
</tr>
<tr>
<td>Third Party Access</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Organisational</td>
<td></td>
</tr>
<tr>
<td>Trusted Sources</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Organizational</td>
<td></td>
</tr>
<tr>
<td>User Access Authorization</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Technical</td>
<td>Legal</td>
</tr>
<tr>
<td>User Access Reviews</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Organizational</td>
<td>Human-oriented</td>
</tr>
<tr>
<td>User Access Revocation</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Human-oriented</td>
<td>Organizational</td>
</tr>
<tr>
<td>User ID Credentials</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Technical</td>
<td></td>
</tr>
<tr>
<td>Utility Programs Access</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/Trustworthiness</td>
<td>Technical</td>
<td></td>
</tr>
<tr>
<td>Infrastructure &amp; Virtualization</td>
<td>IVS-01</td>
<td>IVS-02</td>
<td>IVS-03</td>
<td>IVS-04</td>
<td>IVS-05</td>
<td>IVS-06</td>
</tr>
<tr>
<td>Audit Logging / Intrusion Detection</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Integrity</td>
<td>Confidentiality</td>
<td>Accountability</td>
</tr>
<tr>
<td>Change Detection</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Integrity</td>
<td>Confidentiality</td>
<td>Accountability</td>
</tr>
<tr>
<td>Clock Synchronization</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Organisational</td>
<td></td>
</tr>
<tr>
<td>Information System Documentation</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Availability</td>
<td>Organisational</td>
<td>Legal</td>
</tr>
<tr>
<td>Vulnerability Management</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Availability</td>
<td>Technical</td>
<td></td>
</tr>
<tr>
<td>Network Security</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Confidentiality</td>
<td>Integrity</td>
<td>Privacy</td>
</tr>
<tr>
<td>OS Hardening and Base Controls</td>
<td>IVS-07</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Technical</td>
</tr>
<tr>
<td>Production / Non-Production Environments</td>
<td>IVS-08</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Technical</td>
</tr>
<tr>
<td>Segmentation</td>
<td>IVS-09</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Privacy</td>
<td>Organisational Legal</td>
</tr>
<tr>
<td>VM Security - Data Protection</td>
<td>IVS-10</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Confidentiality Integrity Privacy</td>
<td>Technical</td>
</tr>
<tr>
<td>Hypervisor Hardening</td>
<td>IVS-11</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/ Trustworthiness Availability</td>
<td>Technical</td>
</tr>
<tr>
<td>Wireless Security</td>
<td>IVS-12</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authenticity/ Trustworthiness Availability</td>
<td>Technical</td>
</tr>
<tr>
<td>Network Architecture</td>
<td>IVS-13</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Availability Integrity</td>
<td>Organisational Technical</td>
</tr>
<tr>
<td>Interoperability &amp; Portability</td>
<td>APIs</td>
<td>IPY-01</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability Auditability</td>
</tr>
<tr>
<td>Data Request</td>
<td>IPY-02</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability Auditability</td>
<td>Organizational</td>
</tr>
<tr>
<td>Policy &amp; Legal</td>
<td>IPY-03</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability Auditability</td>
<td>Legal</td>
</tr>
<tr>
<td>Standardized Network Protocols</td>
<td>IPY-04</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability Auditability</td>
<td>Organizational</td>
</tr>
<tr>
<td>Virtualization</td>
<td>IPY-05</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability Auditability</td>
<td>Technical</td>
</tr>
<tr>
<td>Mobile Security</td>
<td>Anti-Malware</td>
<td>MOS-01</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
</tr>
<tr>
<td>Application Stores</td>
<td>MOS-02</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Auditability</td>
<td>Organisational</td>
</tr>
<tr>
<td>Approved Applications</td>
<td>MOS-03</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Organisational</td>
</tr>
<tr>
<td>Approved Software for BYOD</td>
<td>MOS-04</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Auditability</td>
<td>Organisational Human-oriented</td>
</tr>
<tr>
<td>Awareness and Training</td>
<td>MOS-05</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Organisational Human-oriented</td>
</tr>
<tr>
<td>Cloud Based Services</td>
<td>MOS-06</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Auditability Availability</td>
<td>Technical</td>
</tr>
<tr>
<td>Compatibility</td>
<td>MOS-07</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Legal</td>
</tr>
<tr>
<td>Device Eligibility</td>
<td>MOS-08</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Auditability</td>
<td>Organisational</td>
</tr>
<tr>
<td>-------------------</td>
<td>--------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>--------------</td>
<td>----------------</td>
</tr>
<tr>
<td>Device Inventory</td>
<td>MOS-09</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Legal</td>
</tr>
<tr>
<td>Device Management</td>
<td>MOS-10</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Auditability</td>
<td>Technical</td>
</tr>
<tr>
<td>Encryption</td>
<td>MOS-11</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Organisational</td>
</tr>
<tr>
<td>Jailbreaking and Rooting</td>
<td>MOS-12</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Auditability</td>
<td>Organisational</td>
</tr>
<tr>
<td>Legal</td>
<td>MOS-13</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Privacy</td>
<td>Legal</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Accountability</td>
<td>Auditability</td>
</tr>
<tr>
<td>Lockout Screen</td>
<td>MOS-14</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Technical</td>
</tr>
<tr>
<td>Operating Systems</td>
<td>MOS-15</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Auditability</td>
<td>Organisational</td>
</tr>
<tr>
<td>Passwords</td>
<td>MOS-16</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Authentication</td>
<td>Organisational</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Accountability</td>
<td>Auditability</td>
</tr>
<tr>
<td>Policy</td>
<td>MOS-17</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Organisational</td>
</tr>
<tr>
<td>Remote Wipe</td>
<td>MOS-18</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Technical</td>
</tr>
<tr>
<td>Security Patches</td>
<td>MOS-19</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Technical</td>
</tr>
<tr>
<td>Users</td>
<td>MOS-20</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Technical</td>
</tr>
<tr>
<td>Security Incident Management, E-Discovery &amp; Cloud Forensics</td>
<td>SEF-01</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>Organisational</td>
</tr>
<tr>
<td></td>
<td>SEF-02</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>Legal</td>
</tr>
<tr>
<td></td>
<td>SEF-03</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>Organisational</td>
</tr>
<tr>
<td></td>
<td>SEF-04</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>Legal Human-oriented</td>
</tr>
<tr>
<td></td>
<td>SEF-05</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>Organisational</td>
</tr>
<tr>
<td>Supply Chain Management, Transparency and Incident Reporting</td>
<td>STA-01</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Technical</td>
</tr>
<tr>
<td></td>
<td>STA-02</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
<td>Organisational</td>
</tr>
<tr>
<td>Accountability</td>
<td>Network / Infrastructure Services</td>
<td>STA-03</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------</td>
<td>--------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>----------------</td>
</tr>
<tr>
<td></td>
<td>Provider Internal Assessments</td>
<td>STA-04</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability Auditability</td>
</tr>
<tr>
<td></td>
<td>Supply Chain Agreements</td>
<td>STA-05</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Integrity Accountability Confidentiality Authenticity/ Trustworthiness</td>
</tr>
<tr>
<td></td>
<td>Supply Chain Governance Reviews</td>
<td>STA-06</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
</tr>
<tr>
<td></td>
<td>Supply Chain Metrics</td>
<td>STA-07</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
</tr>
<tr>
<td></td>
<td>Third Party Assessment</td>
<td>STA-08</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Accountability</td>
</tr>
<tr>
<td></td>
<td>Third Party Audits</td>
<td>STA-09</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Confidentiality Authenticity/ Trustworthiness Auditability</td>
</tr>
<tr>
<td>Threat &amp; Vulnerability Management</td>
<td>Anti-Virus / Malicious Software</td>
<td>TVM-01</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Availability</td>
</tr>
<tr>
<td></td>
<td>Vulnerability / Patch Management</td>
<td>TVM-02</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Availability</td>
</tr>
<tr>
<td></td>
<td>Mobile Code</td>
<td>TVM-03</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Availability</td>
</tr>
</tbody>
</table>