



Project title: Enforceable Security in the Cloud to Uphold Data Ownership
Project acronym: ESCUDO-CLOUD
Funding scheme: H2020-ICT-2014
Topic: ICT-07-2014
Project duration: January 2015 – December 2017

Ref. Ares(2015)2808558 - 03/07/2015

D1.1 First version of requirements from the use cases

Editors: Ali Sajjad (BT)
Theo Dimitrakos (BT)
Rob Rowlingson (BT)

Reviewers: Neeraj Suri (TUD)
Stefano Paraboschi (UNIBG)

Abstract

This document details the initial security requirements of the four ESCUDO-CLOUD use cases. Each use case addresses a specific application scenario, which can be classified as Infrastructure Provisioning, Cloud Storage, and Cloud Processing scenarios. This permits to cover an entire cloud migration lifecycle where a company may choose to outsource infrastructure, middleware, data and applications. For each use case, we first identify the actors, the actions they can perform, and their goals. Based on their goals, we identify the detailed functional and non-functional requirements of the services central to these use cases. These requirements are determined by the desire of different parties to receive prompt services, and not to have confidentiality, integrity, or availability of their data breached. These requirements are also determined by the legal and business environment surrounding likely deployments. The requirements will be iteratively refined, steering the entire technological development process according to the project goals and objectives.

Type	Identifier	Dissemination	Date
Deliverable	D1.1	Public	2015.06.30



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 644579. This work was supported in part by the Swiss State Secretariat for Education, Research and Innovation (SERI) under contract No 150087. The opinions expressed and arguments employed herein do not necessarily reflect the official views of the European Commission or the Swiss Government.

ESCUDO-CLOUD Consortium

1.	Università degli Studi di Milano	UNIMI	Italy
2.	British Telecom	BT	United Kingdom
3.	EMC Corporation	EMC	Ireland
4.	IBM Research GmbH	IBM	Switzerland
5.	SAP SE	SAP	Germany
6.	Technische Universität Darmstadt	TUD	Germany
7.	Università degli Studi di Bergamo	UNIBG	Italy
8.	Wellness Telecom	WT	Spain

Versions

Version	Date	Description
0.1	2015.05.05	Initial Release
0.2	2015.06.05	Second Release
0.3	2015.06.23	Third Release
1.0	2015.06.30	Final Version

List of Contributors

This document contains contributions from different ESCUDO-CLOUD partners. Contributors for the chapters of this deliverable are presented in the following table.

Chapter	Author(s)
Executive Summary	Rob Rowlingson (BT)
Chapter 1: Introduction	Ali Sajjad (BT), Theo Dimitrakos (BT), Rob Rowlingson (BT)
Chapter 2: Overview of Use Cases	Ali Sajjad (BT), Theo Dimitrakos (BT)
Chapter 3: Elaboration of Use Case 1	Elli Androulaki (IBM), Nikola Knežević (IBM), Christian Cachin (IBM)
Chapter 4: Elaboration of Use Case 2	Florian Kerschbaum (SAP)
Chapter 5: Elaboration of Use Case 3	Ali Sajjad (BT), Theo Dimitrakos (BT)
Chapter 6: Elaboration of Use Case 4	Mercedes Castano Torres (WT), Ignacio Campos Rivera (WT)
Chapter 7: Analysis of Use Cases	Ali Sajjad (BT), Theo Dimitrakos (BT)
Chapter 8: Conclusions	Ali Sajjad (BT), Theo Dimitrakos (BT), Rob Rowlingson (BT)
Chapter 9: Bibliography	Ali Sajjad (BT), Theo Dimitrakos (BT)
Annex A	Ali Sajjad (BT), Theo Dimitrakos (BT)

Contents

1	Introduction.....	14
1.1	Overview of Work Package 1	14
1.2	Purpose of this Deliverable	15
1.3	Use Case Analysis Methodology	16
1.4	Structure of this Deliverable	18
2	Overview of ESCUDO-CLOUD Dimensions and Use Cases.....	20
2.1	ESCUDO-CLOUD Dimensions	20
2.1.1	USE CASE 1: OpenStack Framework (IBM).....	22
2.1.2	USE CASE 2: Secure Enterprise Data Management in the Cloud (SAP)	23
2.1.3	USE CASE 3: Federated Secure Cloud Storage (BT)	25
2.1.4	USE CASE 4: Elastic Cloud Service Provider (WT)	26
2.2	ESCUDO-CLOUD Use Cases	27
2.2.1	USE CASE 1: OpenStack Framework (IBM).....	27
2.2.2	USE CASE 2: Secure Enterprise Data Management in the Cloud (SAP)	28
2.2.3	USE CASE 3: Federated Secure Cloud Storage (BT)	29
2.2.4	USE CASE 4: Elastic Cloud Service Provider (WT)	30
3	Elaboration of ESCUDO-CLOUD Use Case 1 with Security Requirements.....	31
3.1	Introduction.....	31
3.2	High Level “Business” Use Case.....	31
3.2.1	Introductory Scenario.....	31
3.2.2	Purpose	32
3.2.3	Comparison to current practice	33
3.2.4	Objectives.....	34
3.2.5	Stakeholders.....	34
3.2.6	Glossary of Acronyms.....	35
3.3	Use Cases	35
3.3.1	Use Case 1: OpenStack Framework.....	36
3.4	Requirements for relevant Cloud Data Security Solutions	40
3.4.1	Introduction.....	40
3.4.2	Overall Description	41
3.4.3	External Interface Requirements.....	42
3.4.4	System Features	43
3.4.5	System Features	45
3.4.6	Requirements Catalogue	48
3.4.7	Other Non-functional Requirements.....	49
4	Elaboration of USE CASE 2 with Security Requirements.....	51
4.1	Introduction.....	51
4.2	High Level “Business” Use Case.....	51
4.2.1	Introductory Scenario.....	51
4.2.2	Purpose	52

4.2.3	Comparison to current practice	52
4.2.4	Objectives.....	53
4.2.5	Stakeholders.....	53
4.3	Use Cases	54
4.3.1	Use Case: Collaborative Forecasting in Aerospace Fleet Management.....	54
4.3.2	Non-functional Requirements of the use case.....	56
4.4	Requirements for relevant Cloud Data Security Solutions	58
4.4.1	Introduction.....	58
4.4.2	Initial solution architecture in the context of the use-cases.....	59
4.4.3	Overall Description	59
4.4.4	External Interface Requirements.....	60
4.4.5	System Features	61
4.4.6	Requirements Catalogue	64
4.4.7	Other Non-functional Requirements.....	66
5	Elaboration of USE CASE 3 with Security Requirements	67
5.1	Introduction.....	67
5.2	High Level “Business” Use Case.....	68
5.2.1	Introductory Scenario.....	68
5.2.2	Purpose	70
5.2.3	Comparison to current practice	70
5.2.4	Objectives.....	71
5.2.5	Stakeholders.....	71
5.2.6	Glossary of Acronyms.....	72
5.3	Use Cases	72
5.3.1	Use Case: Data Protection as a Service (DPaaS).....	72
5.4	Requirements for Data Protection as a Service use case	84
5.4.1	Introduction.....	84
5.4.2	Initial solution architecture in the context of the use-cases.....	85
5.4.3	Overall Description	85
5.4.4	External Interface Requirements.....	89
5.4.5	System Features	90
5.4.6	Requirements Catalogue	96
5.4.7	Other Non-functional Requirements.....	101
6	Elaboration of USE CASE 4 with Security Requirements	103
6.1	Introduction.....	103
6.2	High Level “Business” Use Case.....	103
6.2.1	Introductory Scenario.....	103
6.2.2	Purpose	105
6.2.3	Comparison to current practice	105
6.2.4	Objectives.....	106
6.2.5	Stakeholders.....	106
6.2.6	Glossary of Acronyms.....	106
6.3	Use Cases	107
6.3.1	Overview	107

6.3.2	Use Case: Elastic Cloud Service Provider	110
6.4	Requirements for Elastic Cloud Service Provider	116
6.4.1	Introduction.....	116
6.4.2	Initial solution architecture in the context of the use-cases.....	117
6.4.3	Overall Description	117
6.4.4	External Interface Requirements.....	119
6.4.5	System Features	120
6.4.6	Requirements Catalogue	125
6.4.7	Other Non-functional Requirements.....	128
7	Analysis of ESCUDO-CLOUD Use Case Relationships and Requirements	130
7.1	Introduction.....	130
7.2	Use Case Relationships	130
7.3	Classification of Requirements	132
7.3.1	Security Properties	132
7.3.2	Access Requirements.....	136
7.3.3	Sharing Requirements	138
7.3.4	Cloud architectures.....	140
8	Conclusions and Next Steps	143
9	Bibliography	145
Annex A:	ESCUDO-CLOUD Integrated Requirements Catalogue	146

List of Figures

Figure 1. Interactions of WP 1 with other WPs of ESCUDO-CLOUD.....	15
Figure 2. Overview of the ESCUDO-CLOUD trust boundaries	21
Figure 3. Overview of the Use Case 1 trust boundaries.....	22
Figure 4. Overview of the Use Case 2 trust boundaries.....	24
Figure 5. Overview of the Use Case 3 trust boundaries.....	25
Figure 6. Overview of the Use Case 4 trust boundaries.....	26
Figure 7. Graphical overview of the IBM use case.....	32
Figure 8. Overview of the relationships between sub-cases in UC 1	36
Figure 9. Sequence of events for Use Case 2.....	56
Figure 10. Trust model for the Use Case 2	59
Figure 11. High level view of the Data Protection as a Service use case	69
Figure 12. Use case diagram for Data Protection as a Service	73
Figure 13. Use case diagram of Cloud Service Store	74
Figure 14. Use case diagram of Cloud Service Store with the Block Storage option	75
Figure 15. Use case diagram of Cloud Service Store with the Object Storage option	77
Figure 16. Use case diagram of Cloud Service Store with the Big Data option.....	78
Figure 17. Sequence diagram for the DPaaS use case	80
Figure 18. Step 1 – Choosing a cloud service provider through a service store.....	81
Figure 19. Step 2 - Select the ‘Data Encryption’ service option.....	82
Figure 20. Step 3 - Provision virtual machine with the Data Protection service	82
Figure 21. Step 4 - Specify a mount point for a data volume that requires encryption	83
Figure 22. Step 5 - Select an encryption key to be associated with the data	83
Figure 23. Step 6 - Configure a security policy to govern the key release	84
Figure 24. Initial solution architecture of the Data Protection as a Service use case to encrypt data volumes attached to virtual machines	86
Figure 25. Introductory scenario for Use Case 4	104
Figure 26. Use Case Diagram (Access* means the privileges of each role of user)	108
Figure 27. UC4 – General architecture	110
Figure 28: Sequence of events with web portal	111
Figure 29. Access through a web browser	112
Figure 30. Access through an agent	112
Figure 31. ESCUDO-CLOUD portal in web browser, user and pwd prompt	113

Figure 32. Page for user listing files and folders accessible	114
Figure 33. User can consult and change the privileges of his files	114
Figure 34. User belong to Writer group or Owner successful opens a file	115
Figure 35. User belong to Writer group or Owner click on “Save” button	115
Figure 36. General Architecture for the access	117
Figure 37. Overview of the Use Cases in the ESCUDO-CLOUD eco-system	131

List of Tables

Table 1. The four dimensions considered in ESCUDO-CLOUD	20
Table 2. Requirements Catalogue for Use Case 1	48
Table 3. Requirements Catalogue for Use Case 2	64
Table 4. Requirements Catalogue for Use Case 3	96
Table 5. Requirements Catalogue for Use Case 4	125
Table 6. Integrated Requirements Catalogue of ESCUDO-CLOUD	146

Executive Summary

This document comprises deliverable D1.1 “First version of requirements from the use cases” in work package 1 of the ESCUDO-CLOUD project. The initial goal of work package 1 in months 1 – 12 is to produce two deliverables to detail selected use cases benefitting from secure cloud services. This deliverable captures the first iteration of this use case analysis and the derived requirements catalogue. D1.2 in month 12 will produce the final version of the security requirements coming from the use cases. The final outcome of work package 1 is a collection of modular, compatible, and interoperable, tools representing the ESCUDO-CLOUD framework implemented in the context of the ESCUDO-CLOUD use cases and meeting their specific requirements and business objectives.

The purpose of this document is to detail the initial security requirements of the four ESCUDO-CLOUD use cases from the project Description of Work (DoW). Each use case represents a real-world problem where one of the industrial partners and the SME participating in the project is involved, and for which there are open security problems that cannot be addressed with current technology. The use cases also cover diverse application domains and provide scenarios able to promote the innovations expected from ESCUDO-CLOUD. Each case stresses specific security aspects, enabling focus and deployment of specific ESCUDO-CLOUD solutions, the implementation of such solutions in a context that is specific to the use-case, as well as exploitation in specific business and usage scenarios. In summary they are:

USE CASE 1: This comprises an OpenStack Framework relating to a cloud-storage platform, which supports server-side encryption with flexible key-management solutions, to be used with OpenStack Swift.

USE CASE 2: This covers secure enterprise data management in the cloud, specifically outsourcing of supply chain interactions in the aerospace engine maintenance industry, based on encrypted database technology.

USE CASE 3: This use case considers the application of data protection as a service via a cloud service store that enables customers to protect their data stored on multi-cloud environments including federated secure cloud storage.

USE CASE 4: This use case considers cloud service brokers or intermediaries offering a secure cloud data storage capability to their customers while possibly leveraging other cloud providers for storing this data and ensuring that data is protected from such other cloud providers and other users.

The use cases are described in detail along with how they relate to four different dimensions that help provide data owners with the combination of security and flexibility when trying to outsource their data resources to the cloud. Each use case provides a description of a scenario in which the use case operates and provides background information, illustrates the problems to be addressed, and discusses why ESCUDO-CLOUD is relevant in the considered scenario. In addition we consider its complementarity with respect to other use cases. Each use case addresses a specific application scenario, which can be classified as Infrastructure Provisioning, Cloud Storage, and Cloud Processing scenarios. This allows the project to cover an entire cloud migration lifecycle where a company may choose to outsource infrastructure, middleware, data and applications.

The analysis of the four use cases follows an agreed methodology. This is based on a combination of user-oriented requirements analysis plus a reasonably detailed declarative presentation of cloud security and/or data protection requirements. The methodology results in an itemized list of detailed functional requirements associated with a range of implementation features. The requirements list is characterised using the four dimensions mentioned above and common requirements extracted for simplification. The amalgamated list is provided as an annex. The initial functional and non-functional requirements obtained as a result of the exercise carried out in this deliverable will be used to drive the research and development work in the technical core work packages WP2, WP3 and WP4. These requirements will ensure that the work carried out in technical tasks of the ESCUDO-CLOUD project corresponds to actual needs of the use cases and enables direct exploitation of that work by the industrial partners.

1 Introduction

ESCUDO-CLOUD is an industry-driven initiative demonstrated through actual business use cases to empower the trust and quality of end-users in order to enable the increased adoption of cloud technology. The objectives of the project are aligned with the stated goals of the Commission: to enable and facilitate a faster adoption of cloud computing throughout all sectors of the economy, to boost productivity, growth and jobs, as defined in the European Cloud Computing Strategy “Unleashing the Potential of Cloud Computing in Europe”.

To meet this objective ESCUDO-CLOUD provides usable techniques enabling data owners (from large companies to individual users) to use the services of cloud service providers (CSP) for storing, managing or processing data in the cloud while enjoying security and privacy. Such protection guarantees are also offered to data owners as well as end-users accessing such data and services. ESCUDO-CLOUD will therefore significantly improve the quality of user experiences in the cloud, increasing social acceptance with trust and assurance. It will remove the possible barrier to the use of CSPs represented by the perceived loss of control over data and of being exposed when using such services, thus enabling their adoption even in scenarios where data owners and users would have felt reluctant. The technology provided will allow owners and users to assess the protection guarantees enjoyed, providing means to assess trust and protection, including auditability. All the techniques will be designed with explicit adoption of open designs, to help to convince users that they can rely on the offered protection.

1.1 Overview of Work Package 1

ESCUDO-CLOUD considers four use cases corresponding to real problems and market strategies of major providers of cloud technologies and services. The work that will be carried out in the project, and the techniques that will be produced, start from the analysis of such use cases and foresee deployment in the use cases. The industrial partners and the SME have clear exploitation plans for managing the innovation brought by the project, which will significantly enhance their offer and will provide actual impact.

This work package is devoted to the use cases considered by the project. It provides requirements (T1.1) which will also drive the research and development work in other WPs, which will be continuously monitored (T1.2), and then deployment (T1.3) and validation (T1.4) of the developed solutions. It ensures the work in the project responds to actual needs of the use cases and enables direct exploitation by the industrial partners. Its goals are to:

- Provide requirements from the different use cases;

- Monitor research activity in the other technical core WPs to ensure the work is in line with the results expected by the use cases; and
- Provide deployment of the project findings and their validation.

The following diagram in Figure 1 presents an overview of the interactions between WP1 and the other WPs in ESCUDO-CLOUD.

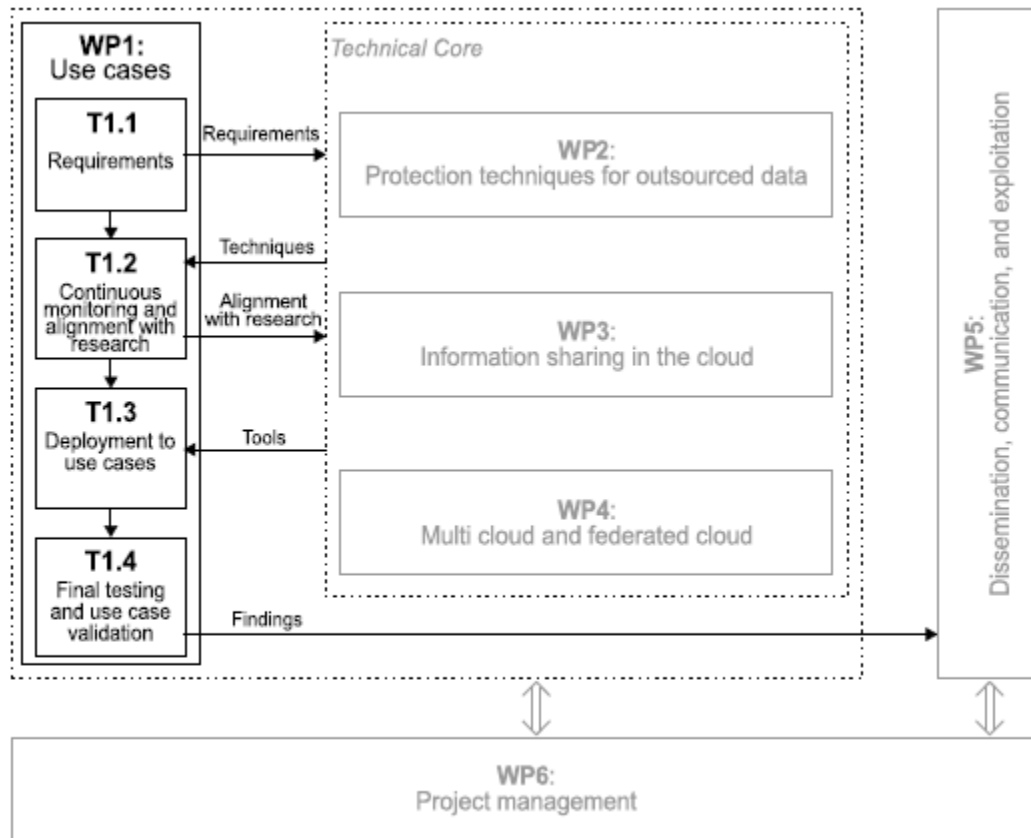


Figure 1. Interactions of WP 1 with other WPs of ESCUDO-CLOUD

The final outcome of WP1 is a collection of modular, compatible, and interoperable, tools representing the ESCUDO-CLOUD framework implemented in the context of the ESCUDO-CLOUD use cases and meeting their specific requirements and business objectives.

1.2 Purpose of this Deliverable

The initial goal of work package 1 in months 1 – 12 is to produce two deliverables to detail the selected use cases benefitting from cloud services. Each use case addresses a specific application scenario, which can be classified as Cloud Storage, Cloud Processing and Infrastructure Provisioning scenarios. This document (D1.1) is the first output of T1.1 that informs the technical core WPs in terms of the ESCUDO-CLOUD use-case requirements.

For each use case, this work will identify the actors, the actions they can perform, and their goals. Based on their goals, it describes the detailed functional and non-functional requirements of the services central to these use cases. These requirements are determined by the desire of different parties to receive prompt services, and not to have confidentiality, integrity, or availability of their data breached. Requirements are also determined by the legal and business environment surrounding likely deployments. The requirements will be iteratively refined, steering the entire technological development process according to the project goals and objectives.

This deliverable captures the first iteration of this use case analysis and the derived requirements catalogue. D1.2 in month 12 will produce the final version of the security requirements coming from the use cases.

1.3 Use Case Analysis Methodology

A project-wide (or "business level") use case is the overall usage described in the project's Description of Work (where 4 are described). A use case describes how a type of user (an actor) uses a system to achieve a goal. A use case also provides a description of a scenario in which the use case operates. A scenario provides the background/context for a use case or set of use cases – it should be closely related to the experiment description. A user story shows the steps and actions a user will take in this use case. A storyboard, in this context, is a graphic description in the form of illustrations or images that show a sequence of events in the user story. They help users and developers to visualise how the user interacts within the use case.

The methodology for use case analysis in this document starts with a description of the business scenario. The scenario provides an overview to put the use cases in context. They describe a representative scenario which covers the key elements of the problem being addressed. The purpose is to describe the experiment in a way that shows the benefits for the end user as if they were able to use a full blown solution. The scenario covers, at a high level, the core features of the project which will be implemented / validated during the use case.

The scenario description can be elaborated in storyboards and "architectural" use case descriptions. These descriptions are complemented by an understanding of each stakeholder involved in the system. The analysis is based on the role the stakeholder plays in the experiment. It should also focus on any particular goals and objectives the stakeholder has. Typically these are captured in a diagram to describe the relationships between them and the different use cases. This allows a check that all the requirements of a use case have been identified by checking that all the identified stakeholders are satisfied with the use case.

The methodology also covers a description the entities which will interact with the system. They can be people or other parts of the system. For example an ATM machine (the system) may have to contact the card holder's bank (in this case an actor). Some actors may be common to more than one use case. In some cases an actor maps well to a stakeholder in the project and this is noted.

In order to understand the significance of the various components of the user story and how they implement or add value to the use case we use the MoSCoW notation [8].

- MUST have this.
- SHOULD have this if possible.
- COULD have this if we have the time.
- WOULD like to have this in the future (but won't do it now).

The methodology also recommends the definition of pre-conditions and post –conditions. Pre-conditions are things which are assumed to be true during the operation of the use case. Post-conditions describe the expected state of the system after the use case has finished. Again this helps in checking the operation of the use case and in ensuring the implementation makes the right assumptions about its operating parameters and outputs.

The analysis also requires a description of a sequence of events. This will take the form of a written description accompanied by a diagram showing the interactions between the actors and the system. The use case should be written in terms which the end user understands. This section makes up the bulk of the use case describes it in detail. Each step in the procedure should be clearly identified. It is intended to flow similar to describing a story e.g. for a demo, teaching or a marketing story.

A storyboard can take this further and is an effective method of prototyping a new system. It allows users and developers to explore the type of user interactions which are required in a new or enhanced system. It shows how an eventual solution could look. A storyboard can take the form of screenshot mock ups or simply drawings indicating how the user interface might look. It should also contain notes which describe any important details which have arisen from creating the storyboard. For example, a storyboard that showed a dialog for entering account details would probably have some notes on the format of account numbers and restrictions on names and addresses.

Using these aspects of the methodology allows initial ideas to be refined and explored allowing finer details to be confirmed with users. They work alongside other requirements gathering techniques and are a valuable way of investigating some of the details.

The above use case analysis based on user oriented requirements can be complemented by a declarative approach to software and system specification. This will consider some of the above factors but also:

- the context and origin of the product being specified: For example, whether this solution is a follow-on member of a product family, a replacement for certain existing systems, or a new, self-contained product. If the section defines a component of a larger system, it relates the requirements of the larger system to the functionality of this software and identifies interfaces between the two. A simple diagram that shows the major components of the overall system, subsystem interconnections, and external interfaces can be helpful;
- the major functions the security sub-system must perform or must let the user perform: A picture of the major groups of related requirements and how they relate, such as a top level data flow diagram or object class diagram, is often effective;

- foreseeable design and implementation constraints that will limit the options available to the developers are highlighted: These might include: corporate or regulatory policies; hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; communications protocols; security considerations; design conventions or programming standards;
- any dependencies the software has on external factors, such as software components that may be reused from another project;
- further identification of the various actors or user classes that will use this software: Actors and user classes may be differentiated based on frequency of use, subset of product functions used, technical expertise, security or privilege levels, educational level, or experience.

Further details such as the operating environment, hardware, software and communication interfaces can be specified in more detail but are not generally appropriate for this early stage of requirements analysis, but can be added in the forthcoming iteration deliverable D1.2.

The methodology then leads to a set of functional requirements by system features, the major services provided by the product. This analysis may be organised by use case, mode of operation, user class, object class, functional hierarchy, or combinations of these, whatever makes the most logical sense for this development. It should provide a short description of the feature and indicate whether it is of High, Medium, or Low priority. It can also include specific priority component ratings, such as benefit, penalty, cost, and risk.

It can cover (if not already clear from previous descriptions and the storyboard) the sequences of user actions and system responses that stimulate the behaviour defined for this feature. These will correspond to the dialog elements associated with use cases.

The functional requirements are clearly itemized and associated with a feature. These are the software capabilities that must be present in order for the user or system to carry out the services provided by the feature, or to execute the use case. Requirements should be concise, complete, unambiguous, verifiable, and necessary. Each requirement is uniquely identified with a sequence number or a meaningful tag allowing an overall requirements catalogue to be derived. This is provided as an annex.

1.4 Structure of this Deliverable

The rest of this deliverable comprises the following chapters:

Chapter 2 provides an overview of each of the use case their relevance to various ESCUDO-CLOUD dimensions and illustrates the complementarity of the use cases.

Chapters 3 - 6 cover each of the four use cases in detail and provides in depth analysis and requirements captured using the methodology described in Chapter 1.

Chapter 7 provides an integration, initial synthesis and characterisation of the requirements and their classification according to ESCUDO-CLOUD dimensions.

Chapter 8 provides the conclusions from this work and makes recommendations for the next iteration of this document in D1.2

Annex A Provides an exhaustive requirements catalogue derived from the four use cases.

2 Overview of ESCUDO-CLOUD

Dimensions and Use Cases

2.1 ESCUDO-CLOUD Dimensions

As it has been observed in the case of Internet communication, the move to encrypt all traffic and data is ubiquitous and inevitable. For example, almost all websites can nowadays be accessed over SSL/TLS-protected connections; email is transported between mail relays through encrypted links, and so on. The same has already happened in the (physical) storage infrastructure market at the storage-area network (SAN) level and at the file-system level; encryption solutions for these services are readily available from many vendors.

Due to the different division of administrative control in cloud computing, data encryption for cloud storage requires some rethinking of the architecture, in terms of encryption placement and key-management options. ESCUDO-CLOUD aims at advancing the technologies to support ubiquitous and user centric data encryption for data stored on the cloud. Offering standards-compatible and regulation-compliant data encryption is a critical feature in future cloud storage services.

ESCUDO-CLOUD considers four different dimensions [6] that help provide data owners with the combination of security and flexibility when trying to outsource their data resources to the cloud. These dimensions are shown in Table 1.

Table 1. The four dimensions considered in ESCUDO-CLOUD

Security properties	<i>i) Confidentiality; ii) Integrity; iii) Availability</i>
Access requirements	<i>i) Upload/download; ii) Fine-grained retrieval; iii) Write operations</i>
Sharing requirements	<i>i) Access by data owners; ii) Selective sharing with other users/owners</i>
Cloud architectures	<i>i) Single cloud provider; ii) Multi clouds and federated clouds</i>

- **Security properties.** We distinguish the different security properties that may be required to be provided. Intuitively, security properties refer to the classical CIA paradigm: Confidentiality, Integrity, and Availability, where this latter can be interpreted in the cloud scenario as satisfaction of service-level guarantees promised by the CSPs (usually expressed via Service Level Agreements).
- **Access requirements.** Here we distinguish three kinds of scenarios: i) access to data requires a primitive upload/download, and protection refers therefore to data-at-rest; ii) access to data requires fine-grained data retrieval and execution of queries (i.e., users are interested in

retrieving specific portions of the data that satisfy certain selection criteria), and protection refers therefore to the protection of also computations and query results; and iii) access to data entails both access retrieval and enforcement of write operations (by the owner and/or other authorised users), and protection refers to the protection of the actions as well as their effects on the data.

- **Sharing requirements.** Here we distinguish: i) a simple scenario where a data owner relies on the cloud for enjoying external storage for her own use and access; and ii) a more complex scenario where the data owner can rely on external storage for making her data available to others, and disseminating and sharing them in a selective way.
- **Cloud architectures.** Here we distinguish two scenarios. The first one considers a single CSP, entailing all the problems of guaranteeing data owners the control on their data, with the ability of fine-grained retrieval and sharing. The second extends to the consideration of multiple clouds and federated clouds, empowering data owners with the ability of selectively relying on different CSPs, depending on trust and economic factors, leveraging different CSPs for security, and operating with federations of objects.

These ESCUDO-CLOUD dimensions, with their different configurations, correspond to different scenarios and challenges that are addressed by the use cases described shortly. However, the common theme is to provide security by wrapping the data with a protection layer; this operates with associated metadata and access methods that enable fine-grained data retrieval, access support, and selective sharing to the owner and authorised users, while protecting data and actions on them from the untrusted cloud service providers (CSP). An overview of this is given in Figure 2, emphasizing the system trust boundaries.

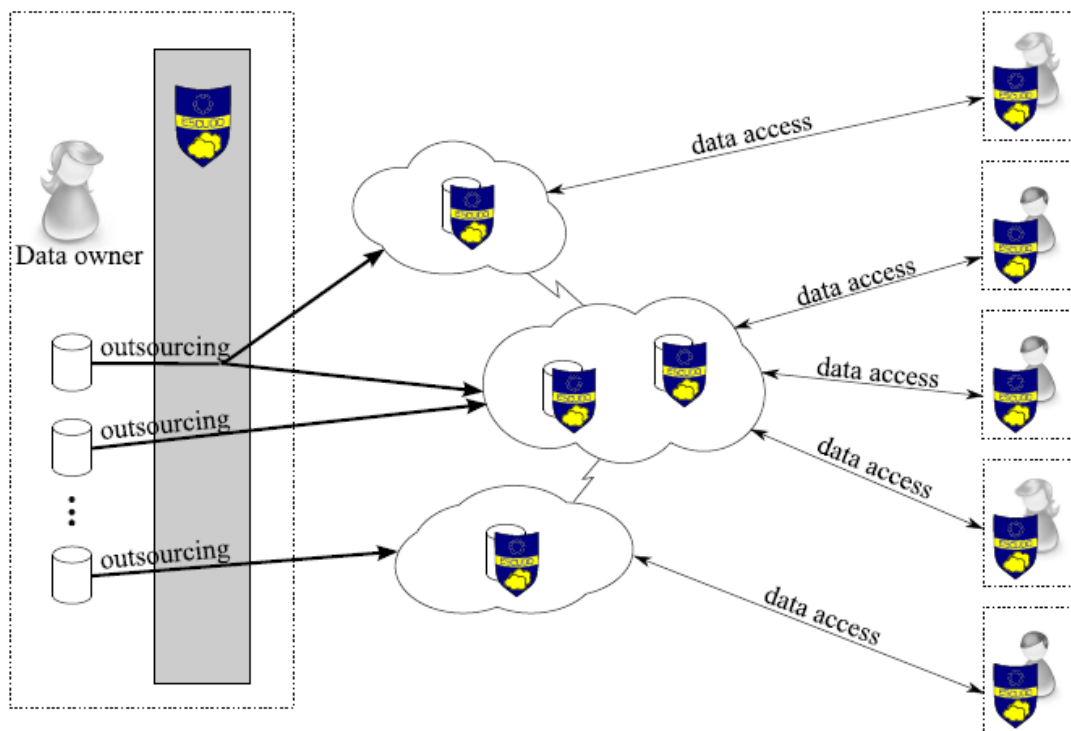


Figure 2. Overview of the ESCUDO-CLOUD trust boundaries

The relevance of these four dimensions of ESCUDO-CLOUD to each use case scenario is detailed as follows:

2.1.1 USE CASE 1: OpenStack Framework (IBM)

The OpenStack data-at-rest encryption use case, which is driven by IBM, provides an excellent opportunity to demonstrate some of the key technologies developed by ESCUDO-CLOUD. From Work Package 2, Task 2.1 (Protection of data at rest), and Task 2.2 (Key-management solutions), the main expected results will consist of technologies to protect the confidentiality and the authenticity of the stored data. In particular, methods for encryption and integrity verification will be applied in the context of the OpenStack cloud platform and demonstrate how the methods of ESCUDO-CLOUD can lead to more security in cloud platforms.

Furthermore, the secure deletion of data will be supported and demonstrated in the OpenStack cloud storage platform. This technology will highlight one of the main features offered by the ESCUDO-CLOUD key management approach. Figure 3 shows the overview of the Use Case 1 trust boundaries, with only the data owner being the trusted entity and the data is encrypted at rest a single OpenStack based cloud service provider.

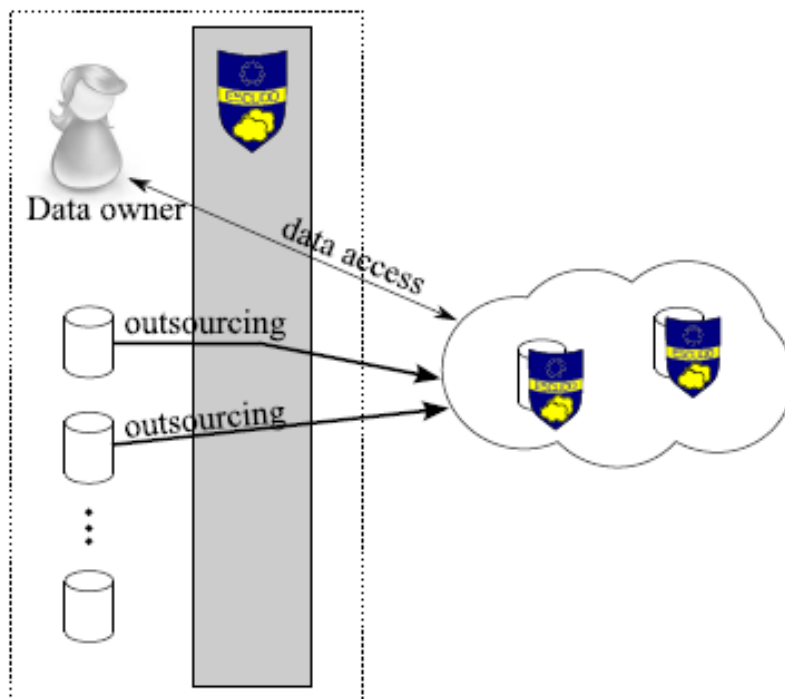


Figure 3. Overview of the Use Case 1 trust boundaries

From Work Package 3 (Information sharing in the cloud), Task 3.2 (Secure multi-user interactions and sharing), the technology to guarantee integrity and consistency of data accessed by multiple users will be demonstrated through OpenStack cloud storage. This novel technology protects data that is shared and concurrently accessed by multiple clients from being altered or modified. Showing this enhancement in the OpenStack cloud storage platform represents a key step to promoting and exploiting this result.

Taken together, the technology demonstrated in use case 1 aims at the business goals of protecting against insider attacks, ensuring compliance with legal regulations, and increasing the value of the storage service in response to customer demand. This directly supports the main approach of ESCUDO-CLOUD.

2.1.2 USE CASE 2: Secure Enterprise Data Management in the Cloud (SAP)

This aerospace engine MRO use case is particularly suited for ESCUDO-CLOUD. On the one hand there is a clear data sharing requirement best managed in the cloud and on the other hand the data to be exchanged is highly sensitive, maybe even personally identifiable.

We have to consider the following data sharing risks:

- *Loss of information advantage.* Disclosing sales information to other organizations may change relationships among partners in a supply chain, since it may reduce the “information rents” effect for which especially a weak partner profit;
- *Reconstruction of strategic decisions.* Sharing of data increases the visibility of operations for all companies involved; thus if confidentiality is not preserved, a competitor could anticipate company’s future plans;
- *Development of a competitive product/service.* With the knowledge of suppliers, competitors can develop new competitive products or offer more competitive services;
- *Weakening of the bargaining power after disclosure of purchase or supply volume.* A customer may compare its own purchase volume to that of other customers in order to calculate its share; such information can be used to strengthen its bargaining power. At the same way, suppliers can calculate their share of overall supply; this information increases their power over the customer in a business negotiation.

ESCUDO-CLOUD is ideally positioned to help overcome these risks. Using ESCUDO-CLOUD technology, e.g. developed in the context of WP3, allows data protection in the cloud while still enabling data sharing. These are the types of solutions required for the future adoption of the use case. Figure 4 shows the overview of the Use Case 2 trust boundaries, with the data owner and users being the trusted entities and the data is encrypted in a database on a single cloud service provider.

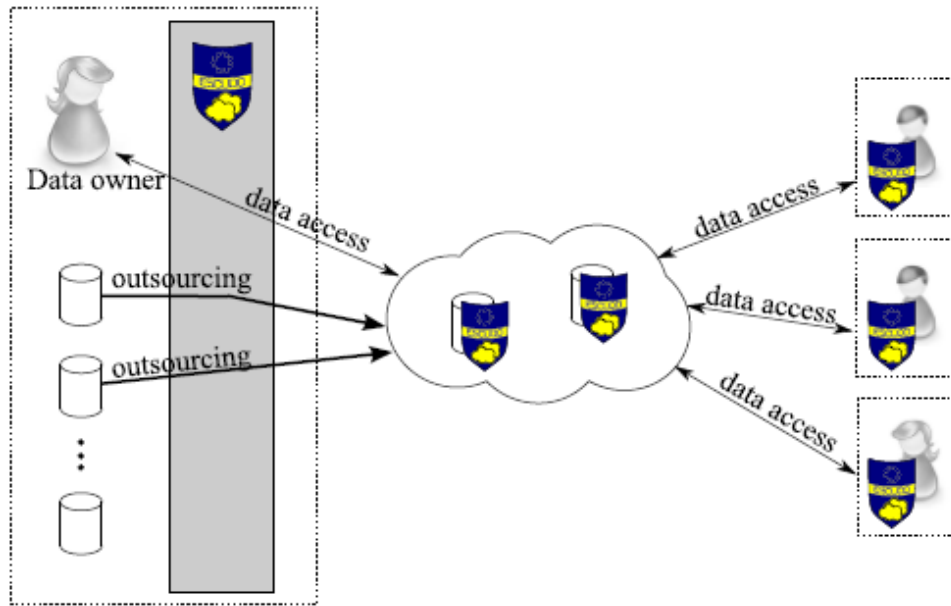


Figure 4. Overview of the Use Case 2 trust boundaries

The main technical functionality of this use case is the computation of the forecasts for the aggregated overhaul demand. The process can be broken down in two major steps. In the first step the individual demand forecast is estimated. The calculation for this is to be applied on the encrypted usage data of each customer. The second step in the process is the secure aggregation of the interim results. We want to compute a sum which would be straightforward if the interim results were centrally available in plaintext. They are, however, sensitive data for each customer and we require a secure aggregation protocol across different keys.

To grant or restrict shared data access to MROs and airlines processing unencrypted query results, data owners have to implement additional fine-grained access control mechanisms. Implementing such a multi-user mode using encrypted query processing for a single user operating with one key combined with an additional authorization step at the application server can be compromised: Assume that a user working for the data owner and a service provider's employee collude. If the user knows the decryption key of the data and the employee provides the encrypted data stored in the database, they are able to decrypt all data bypassing the access control mechanisms.

For use case 2 the following ESCUDO-CLOUD dimensions are important and can be further detailed:-

- Security properties: Confidentiality and integrity of access control decisions are very important. An access decision should not be delegated to the cloud provider.
- Sharing requirements: Data is supposed to be shared in a selective manner with others.
- Access requirements: Fine-grained access as in a relational database system and data may be added, but also sometimes updated or deleted.
- Cloud architectures: The use case is initially targeted for a single cloud architecture as found in most business application.

2.1.3 USE CASE 3: Federated Secure Cloud Storage (BT)

The core deliverable of the Data Protection as a Service use case is to ensure the confidentiality, integrity and availability of the customer's data, which is enforced by using client-side encryption approach. In addition to this core responsibility, the access and usage of the key management and access control features is also directly relevant to the ESCUDO-CLOUD security properties. The key challenge addressed by ESCUDO-CLOUD here is to offer the key management feature and the policy-based access control feature as a service through a cloud service store. The instance of a key management service and the access control service have to be tightly coupled for each customer, which will allow them to specify key release rules that are applicable only under specific conditions. Figure 5 shows the overview of the Use Case 3 trust boundaries, with the only the data owner being the trusted entity and the data is encrypted on different types of storage medium on multiple cloud service providers.

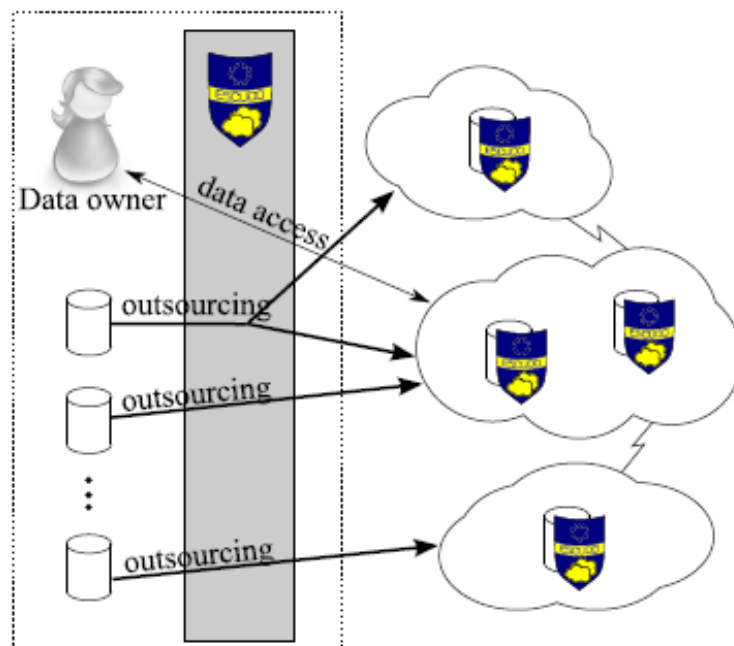


Figure 5. Overview of the Use Case 3 trust boundaries

Furthermore, two of the sharing requirements of ESCUDO-CLOUD have to be catered for by this use case. The first is the simpler situation where only the customers have access to the encryption keys and thus no one else, especially the cloud service providers, has the ability to read the plaintext data. The second is a more complex situation where the customers can make their data available to others by sharing the keys and modifying access control policies.

It is also relevant from the perspective of ESCUDO-CLOUD access requirements, as it has to provide the customers with the means to access and retrieve their data from different types of cloud-based storage services, e.g., block storage, object storage and Big Data services. This also includes the domain aspects of controlling access to the data through policy based security rules that enforce the release of keys.

The last point of relevance to the ESCUDO-CLOUD effort is the operation of the use case in a multi-cloud environment, providing the ability of selectively using different cloud service providers depending on their choice.

2.1.4 USE CASE 4: Elastic Cloud Service Provider (WT)

The current typical usage of cloud services involves continuous operations on sensitive data. Though data at rest may be stored encrypted (and therefore securely), when it is operated on, it must be decrypted. This constant encryption and decryption of user data means that the decryption key is present in RAM somewhere. It may be in the OS, it may be in the Data Base Management System (DBMS), or it may even be in the application itself. This means that a user with access to a system inside the CSP can access the database decryption key, or potentially even the unencrypted database contents, from the RAM, or ‘working memory,’ of the computer. As a result, the robustness of the database encryption scheme becomes nearly irrelevant and would likely not have posed a substantial barrier to someone with the capability to circumvent authentication protocols in the first place. Figure 6 shows the overview of the Use Case 4 trust boundaries, with the only the data owner and the data users are the trusted entities and the data is encrypted on file storage services on multiple cloud service providers.

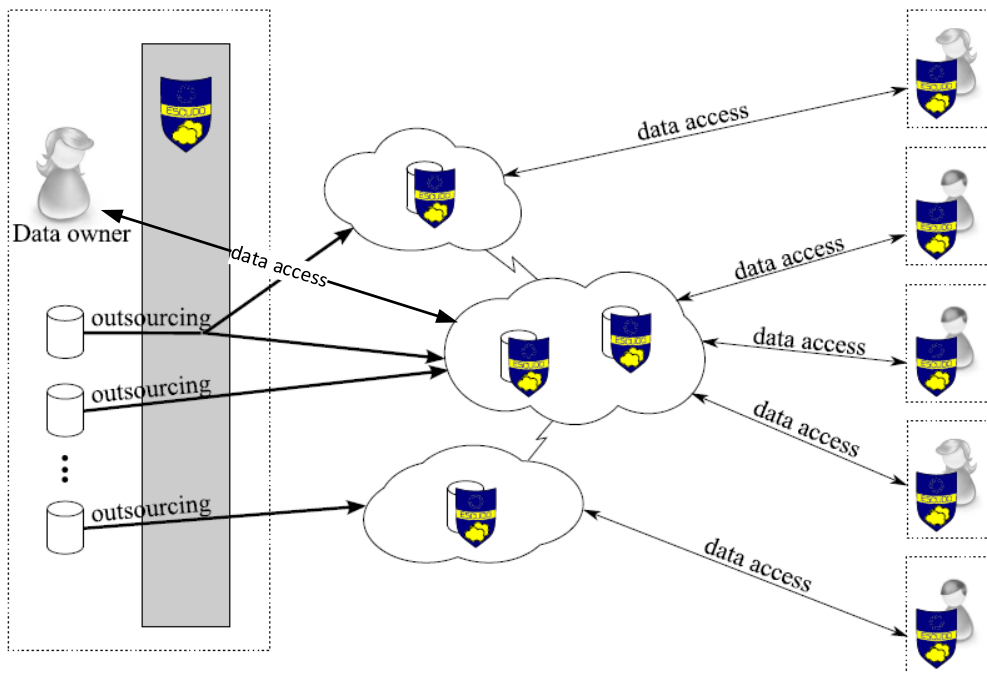


Figure 6. Overview of the Use Case 4 trust boundaries

Opinions vary on how and where to use encryption in the cloud. Regardless of the approach, key management remains a challenge as businesses walk a fine line between trust and control between their own organization and the cloud service provider. Some CSPs combine powerful data encryption with patented homomorphic split-key encryption technology to increase security and protect keys. One part of the key is hosted by the CSP, while the second part – the master key – is held by the customer. The result is that customers control their data and do not need to trust anyone

else with their keys. This technology easily encrypts any disk or data storage unit with proven encryption algorithms such as AES-256 and makes it safe from hackers, unauthorized access, competitors, and other threats.

ESCUDO-CLOUD not only encrypts all data files stored on the CSP, but also offers a uniform user interface, irrespective of what CSP is being used. Furthermore, the middleware will be responsible of the decryption of the data files when the user with the correct privileges access to them (through a web browser or with an agent).

2.2 ESCUDO-CLOUD Use Cases

The validation of the security techniques and tools developed in ESCUDO-CLOUD will be performed through their deployment to four use cases. Each use case represents a real-world problem where one of the industrial partners and the SME participating in the project will be addressing open security problems that cannot be addressed with the current technology. The use cases also cover diverse application domains and provide scenarios able to promote the innovations expected from ESCUDO-CLOUD. All the use cases have common aspects since in each of them the use of cloud services naturally introduces the spectrum of security problems tackled in ESCUDO-CLOUD. However, each of them stresses specific security aspects, enabling focus and deployment of specific ESCUDO-CLOUD solutions, as well as exploitation in specific scenarios.

In the following sections, we provide a high level summary for each use case. Also, we provide a characterisation of the use case in terms of its complementarity with respect to other use cases, with the indication of the main industry partners involved in its deployment and validation.

2.2.1 USE CASE 1: OpenStack Framework (IBM)

The scenario of this use case relates to a cloud-storage platform, which supports server-side encryption with flexible key-management solutions. As such this use-case is particularly applicable for the development of internal cloud solutions as well as for CSPs building private or public cloud solutions using open source frameworks such as OpenStack. In particular, it consists of data-at-rest encryption and key management solutions, to be used with OpenStack Swift, an object-storage system that runs on commodity hardware and provides failure resilience, scalability, and high throughput in software. Encryption occurs on the server side under the governance of the storage provider; encryption inside the storage platform is an important feature for large-scale and enterprise-class storage systems.

Coupled with a suitable key-management solution that is able to control and securely erase cryptographic keys, the encryption technology also supports the controlled destruction of data, which is called secure data deletion here. Data-at-rest encryption and secure deletion are important requirements for enterprise-level cloud storage services.

The goal of this use case consists of adding cryptographic protection technology inside a private cloud platform, in particular, to storage systems. Clients of cloud services and operators of the CSPs

benefit from data encryption in the storage systems, so as to make the system resistant to attacks that target lower layers of the infrastructure.

A wide subset of IBM's storage and cloud storage offerings and products make use of the OpenStack framework. This framework does not yet have some of the required enterprise-grade features. Thus, applying this technology to extend the OpenStack framework with cryptographic protection, and to offer enterprise-grade services to its customers is an important competitive feature in the market.

2.2.2 USE CASE 2: Secure Enterprise Data Management in the Cloud (SAP)

SAP considers the specific use case of outsourcing the supply chain interactions in the aerospace engine maintenance industry. So called, maintenance, repair and overhaul (MRO) providers offer their services to several airlines leveraging cost savings by streamlining the process. In general, two main business-optimizing services have to be guaranteed in the aero engine overhaul supply chain: the collaborative demand forecasting (CF), and the collaborative planning and scheduling (CPS) of the overhaul activities. The first one allows MRO service providers to obtain demand forecasts from all customers based on on-condition engine status observations, reducing so overall costs due to a more accurate capacity planning; while the collaborative planning and scheduling guarantees better supply chain performance, since an ideal receipt point for each engine can be computed.

Traditionally, each party on each stage of the supply chain has its rather isolated forecasting processes which are mainly based on data of historical demand that arose from their direct customers. The problem with these orders from the next stage is that they are again results of an isolated forecast and in general don't match the actual sales on the buyer's stage. Instead, they tend to have a larger variance. This effect of demand distortion results in amplified forecasting numbers and dramatic swings in demand increasing with every step on the supply chain further away from the end customer. This phenomenon is known as the bullwhip effect.

However, this information does exist, and CF is an attempt to bring them together to create a single, more accurate forecast which is accepted by all collaborating members of the supply chain. In a collaborative forecasting process ideally all supply chain members add their particular expertise and information to find the best possible forecast. The information about end customer demand is shared with the upstream supplier, so demand distortion can be reduced drastically. This again will drastically reduce the bullwhip effect.

A central issue for maintenance and support service providers concerns the management of the growing amount of information generated by the development of highly complex aircraft systems and by stakeholders' requirements in terms of dependability increase and Life Support Cost (LSC) decrease. To face these problems, maintenance and support actors are depending more and more on ICT solutions. These are one of the main elements not only to improve the effectiveness and efficiency of the maintenance process for complex systems with a long lifecycle, but also to reduce

the associated risks and to contribute to a more efficient business process. The benefits linked to the use of ICT systems in this business segment are:

- More controlled content sharing;
- Information exchange and knowledge management;
- Coordination of maintenance process with other processes;
- Connection to strategic business objectives and external stakeholder requirements.

The technological challenge of this use case is to develop new supply chain cooperation systems, based on encrypted database technology. This technology is based on the search and aggregation of encrypted data and can be applied in planning the MRO service to different customers without knowing the actual status of the aero fleet nor the capacity usage of the service provider, not the inventory status.

Supply chain collaboration is the alignment of individual plans and strategies of the involved parties. The stronger coordination and the overcoming of information asymmetries with partners shall conduce to improvements of the supply chain performance. Additionally, the uncertainties in demand may be reduced by taking into account interactions at other levels of the supply chain. Better knowledge of the downstream demand enables the customer to facilitate the vendor's predictability skills concerning his production and delivery capacities by providing him with appropriate data. Reduced inventory and hence lower costs are potential benefits for the partners.

2.2.3 USE CASE 3: Federated Secure Cloud Storage (BT)

This use case considers the application of data protection in multi-cloud environments including federated secure cloud storage. It will offer data protection as a service via a cloud service store that enables customers to protect their data stored on multiple cloud platforms. As such this use case is particularly applicable for Managed Security Service Providers and for Cloud user organisations who want to control the protection of data at rest across multiple cloud environments and apply uniform data protection and data access policies for heterogeneous data stored in a multiplicity of cloud providers (including internal, private and public clouds).

The data is protected by leveraging a cloud-based data protection service for encrypting independently of the cloud provider hosting it, and ensuring that cloud service providers have no access to the encryption keys or its protection and access control policies. The encrypted data can be stored on multiple cloud storage services like block storage, object stores and Big Data clusters. Access control and key management are offered as tightly coupled services that manage the protection of the data via an integrated policy framework. This tight integration will ensure that the decryption of the protected data is only possible in the client's environment following a policy-based approval procedure and the resulting release of the encryption key. The encryption and decryption process will be transparent to applications and end-users while the data-at-rest will always stay in encrypted state on the multiple cloud platforms, in compliance with specific data security standards and regulations.

2.2.4 USE CASE 4: Elastic Cloud Service Provider (WT)

This use case considers the use of an elastic cloud service provider. As such this use-case is of particular relevance for Cloud service brokers or intermediaries offering a secure cloud data storage capability to their customers while possibly leveraging other cloud providers for storing this data and ensuring that data is protected from such other cloud providers and other users.

In this use case scenario a data owner user has access to his/her files stored in the cloud using some middleware available on the client (web portal or agent). The middleware will enable the communication between the end user and the CSP. Furthermore, by using an elastic cloud, it will be possible to adapt the capacity of the cloud to the requirements of the user. Furthermore, by using the ESCUDO-CLOUD middleware with an elastic cloud, the encryption provided by ESCUDO-CLOUD will be present in the third party cloud too (so, ESCUDO-CLOUD will be present for all data transfers).

With the ESCUDO-CLOUD middleware, the users will have secure access to their data hosted by the cloud providers such that the data is not compromised. They will also be able to manage their data from a web browser or an agent (installed in their devices). Finally, it will be possible to synchronize stored files from third-parties like Google Drive or Dropbox. So, by using ESCUDO-CLOUD the users will be able to leverage the cloud services that they would typically request from a CSP but with a higher level of security, access control and assurance.

The techniques researched and developed by ESCUDO-CLOUD will be applicable in many paradigms of data protection in the cloud service, as demonstrated by the above use cases, although their realisation may take the form of diverse implementations resulting in different solutions addressing the specific requirements of different use-cases. For example, the first use-case focuses on enhancing open source cloud platforms with built-in data protection capabilities, while the second use case focuses on the improvement of data protection for cloud-based data-sharing applications and platforms. In both cases the data protection is essentially governed by the cloud infrastructure, platform or data-base service provider. In contrast, the third use-case focuses on protecting data hosted on multiple cloud platforms at the infrastructure or hosted application level while shifting the governance of the data protection and data access management off the cloud infrastructure to the data owner assisted by a security service that is independent of any cloud infrastructure. While the fourth use-case is about an intermediary building an elastic, cloud-based secure storage and data synchronisation service by leveraging 3rd party cloud infrastructures and data storage services while the intermediary assures and governs the data protection independently of the cloud hosts.

3. Elaboration of ESCUDO-CLOUD Use Case 1 with Security Requirements

3.1 Introduction

Problem statement: Encryption of data at rest and secure data deletion are important requirements for enterprise-level cloud storage services. Encryption of data at rest is required to handle the case of stolen or improperly-decommissioned storage media (e.g., disks, tapes). Secure data deletion is required to delete large quantities of data in a time- efficient manner, without relying on overwriting-based techniques – which are ineffective on certain media and vastly inefficient in general – or on the physical destruction of storage media, which permits for very little flexibility when it comes to the selection of which files, objects or volumes are required to be deleted.

Solutions for both requirements are usually obtained with cryptography-based techniques: if data is encrypted before being committed to storage, it cannot be accessed even in case storage media is compromised; if keys are properly managed and carefully destroyed, large subsets of data is no longer accessible.

Contribution: A wide subset of IBM’s storage and cloud storage offerings and products make use of the OpenStack framework. This framework does not have required enterprise-grade features. Thus, we would like to extend the OpenStack framework to offer enterprise-grade services to its customers, through a design for cloud-based encryption of data at rest and secure deletion, over the key-management aspects in the cloud (considering the transactional guarantees offered by cloud-services as opposed to traditional, on-premise services), as these are some of the objectives of ESCUDO-CLOUD.

3.2 High Level “Business” Use Case

3.2.1 Introductory Scenario

Cloud platforms consist of computing infrastructure commonly realized in software, yet, they provide abstractions that have traditionally been available in hardware. Virtual machines, logical block stores, database services, and so on, are cloud-computing equivalents of well-understood physical-world concepts. In cloud platforms these are emulated by software (for example, virtual machines provided by hypervisors). This underlying software platform therefore becomes a

universal infrastructure that serves many different clients ("tenants") and different tasks. As it is used by many tasks and users, clients may not trust the infrastructure to the same extent as they would trust an equivalent service made from a physical resource.

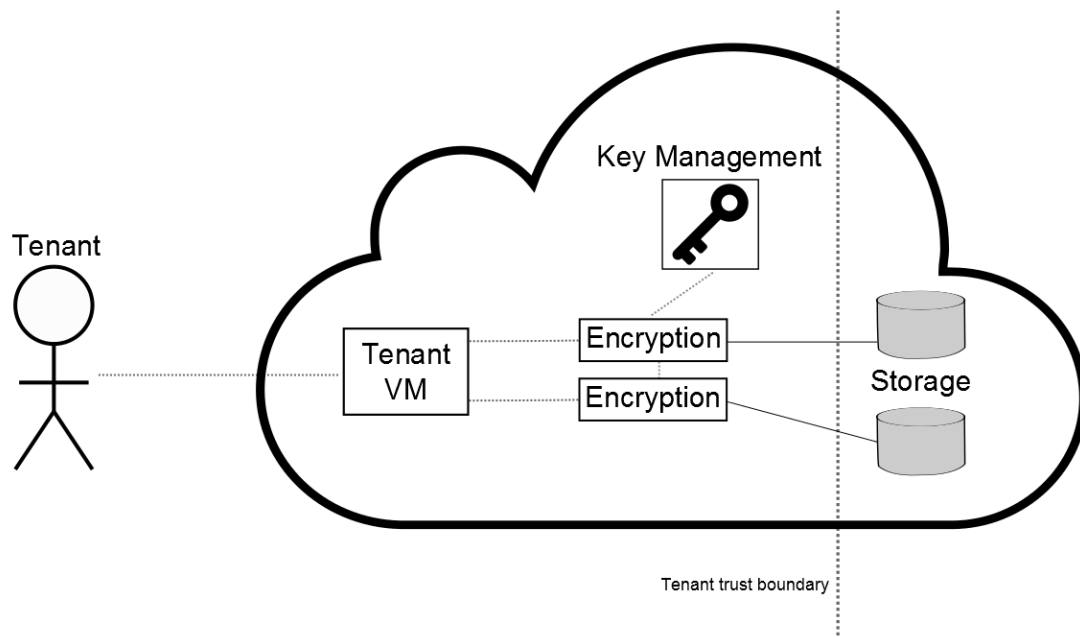


Figure 7. Graphical overview of the IBM use case

The goal of this use case consists in adding cryptographic protection technology inside a cloud platform, in particular, to storage systems, as depicted in Figure 7. Clients of cloud services and operators of the cloud providers benefit from data encryption in the storage systems, so as to make the system resistant to attacks that target lower layers of the infrastructure.

There are multiple business cases connected to this: First, a cloud provider directly benefits from the added security in the infrastructure that it provides. Second, an IT services company may sell a cloud platform to a client, or run a cloud platform on behalf of a large client, such that the client may operate as a (re-)seller of cloud services. Third, an IT company may provide a cloud solution to an enterprise-sized customer, and the customer runs this cloud platform with internal clients, as an easy-to-use service replacing traditional storage and computing systems. Fourth and last, a large company may itself operate a private cloud, which offers storage, compute, and network resources to internal clients. In all four mentioned cases, insider attacks pose a threat that can be mitigated by the technology.

3.2.2 Purpose

In particular, this use case consists of data-at-rest encryption and key management solutions, to be used with OpenStack Swift, an object-storage system that runs on commodity hardware and provides failure resilience, scalability, and high throughput in software. Encryption occurs on the server side under the governance of the storage provider; although this design cannot protect data as

strongly as client-side encryption, encryption inside the storage platform is an important feature for large-scale and enterprise-class storage systems.

This solution aims at the business goals of protecting against insider attacks, ensuring compliance with legal regulations, and increasing the value of the storage service in response to customer demand. Even though encryption keys are ultimately not protected from some high-level security operators of the provider, the clients and the provider benefit from the added security because of the many further actors that pose dangers to the confidentiality and integrity of cloud-stored data.

Naturally, an encryption-based protection solution reduces the security of the data to the security of the encryption keys. Thus, the key-management problem is the main focus of this use case. Many options exist for managing keys, with widely varying security and performance characteristics. Cloud key management may indeed require markedly different solutions, depending on whether its objective is to manage keys for the client, for the cloud provider, or for a mixture of both. In the first case, the client may want to use the resources of the cloud provider to reliably manage its own keys. To further strengthen the security of the solution, implementations may choose to leverage Hardware-Security Modules (HSMs) that physically reside on the premises of the cloud provider, while they are logically used by and trusted by the end user.

A second goal that can be achieved using encryption is the capability to securely delete data from a storage system. This feature lies in the interests of the clients and of the CSP. As modern storage systems contain many layers between the client and the actual physical storage medium, traces of logically removed data are often left behind and can be recovered with little effort. In order to protect against such leakage, a design for policy-based secure deletion of data in the cloud storage platform is envisaged, which relies on the proper erasure of cryptographic keys by a key-management system. Basically, the approach is to choose a fresh key, to re-encrypt the data that remains accessible with the new key, and the data to be deleted is left out from this process. Then the old key is destroyed.

As many regulations demand from providers to erase data, especially to erase personal and sensitive data, this feature is relevant and interesting from the client's perspective, in the sense that clients will pay a premium for this service. Furthermore, it is also in the interest of a cloud provider to offer a secure deletion capability because it limits exposure of data in the internal infrastructure.

3.2.3 Comparison to current practice

The currently existing open-source cloud object storage solutions do not have support for data-at-rest encryption and neither integrates with enterprise-strength key-management systems. In particular, systems based on hardware-security modules (HSMs) are not available in the cloud context, except for one or two isolated vendor-specific solutions. Unfortunately, the protocols which are currently used today to interact with HSMs, such as PKCS#11, are not suited for the dynamic, virtualised and multi-tenant environment of the cloud. Integrating these features with cloud platforms is an important goal.

Furthermore, the highly concurrent nature of the cloud storage system poses challenges for the coordination among the encryption worker processes. A highly concurrent encryption solution is desired that retains the scalability of the storage platform, and yet integrates with the strictly controlled and costly interfaces for key management.

No solution is known on the market currently that offers a fine-grained secure deletion capability. Clients and service providers should be able to specify a secure erasure operation based on attributes attached to stored data, distinguishing data to be retained from data to be destroyed in a flexible way, at the time when data is created. Current solutions would require to scan through the storage system and to re-classify the data for a particular deletion need.

3.2.4 Objectives

- To provide flexible solutions for protecting data stored in cloud platforms through data-at-rest encryption and cryptographic integrity verification.
- To offer encryption features for an object storage platform.
- To manage encryption keys and cryptographic secrets according to the needs of cloud providers and clients, bridging the gap between traditional strong security based on HSMs and the elastic, distributed, and scalable nature of cloud platforms.
- To provide a capability for fine-grained controlled secure deletion of data in storage systems based on re-keying and re-encrypting data and destroying sensitive key material.

3.2.5 Stakeholders

The cloud storage encryption scenario contains three potential stakeholders: the provider, the tenant, and the clients.

A provider offers cloud services to the tenants and to the clients. The provider may internally consist of architects, operators, auditors, and more functions.

The clients are individuals or roles operating under the common umbrella of a tenant. For example, a large company could be a customer of a cloud provider and in that role contract cloud services for many clients, which might be different employees of the company or different organizational units inside the company. The distinction between clients and tenants is sometimes not necessary and all clients of a tenant can be identified with their respective tenant.

Depending on the particular scenario, one may also distinguish many more stakeholders, such as service resellers, equipment suppliers, software-engineering contractors, and so on.

Many particular instantiations of the trust model are possible. Cryptographic keys could reside with the tenants or with the provider, for example, or keys could be stored by the provider but hidden from its operators, as the keys might be entirely controlled by the tenant. A flexible solution will support multiple trust models in this sense.

3.2.6 Glossary of Acronyms

Acronym	Definition
UC	Use Case
HSM	Hardware-Security Module
CRUD	Create, Read, Update, Delete
KM	Key Management
IKM	Infrastructure Key Management
TKM	Tenant Key Management
KMIP	Key Management Interoperability Protocol
PKCS	Public-Key Cryptography Standard

3.3 Use Cases

Figure 8 depicts the relationship between the sub-cases we cover in this Use Case. These sub-cases will be covered in this section.

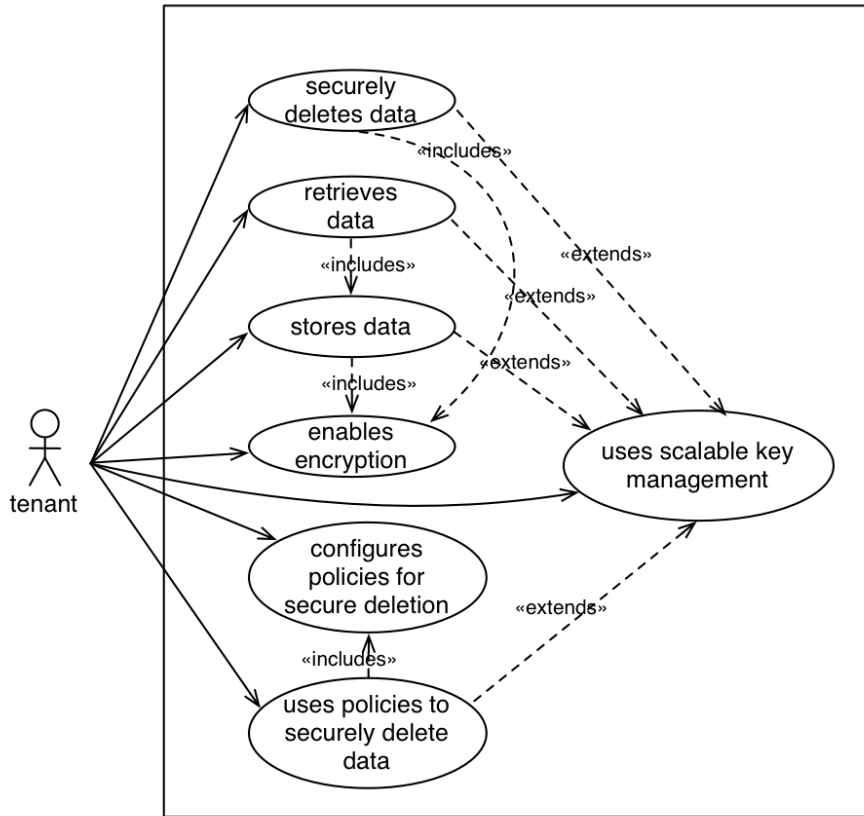


Figure 8. Overview of the relationships between sub-cases in UC 1

Requirements (general):

- An attacker that has only access to the disks that belong to the object storage cannot read the plaintext content

Requirement for secure deletion use cases (irrespective of whether the object store is used for storing keys or data):

- An attacker that has only access to the disks that belong to the object storage cannot recover the content of deleted files
- An attacker that has access to the entire object storage infrastructure cannot recover the content of deleted files.

3.3.1 Use Case 1: OpenStack Framework

We distinguish the following use cases including in OpenStack framework use case:

3.3.1.1 Client enables encryption on object storage

- Actors: Clients and the (private or public) object storage infrastructure, where client-data is stored.
- Purpose: Clients enable encryption on the object store their data reside

- Priority:

MUST: Mechanisms that enable the client to enable or disable encryption in its storage backend

- Pre-conditions:

- Proper client authentication on the storage backend
- Client-specific accounts on the storage backend

- Post-conditions:

- The client is able to enable or disable encryption for the data stored on its storage backend

3.3.1.2 Client leverage an (off-premise) encryption-enabled object storage to store their data

- Actors: Clients and the object storage infrastructure (cloud), where client-data is stored.

- Purpose: Client ability to outsource storage of their data to the cloud.

- Priority:

- On the object storage side:

MUST: Data encryption capability assuming access to a key manager

COULD: Secure and scalable key management

COULD: Stored data integrity protection/check

- On the client side:

COULD: Secure key management

- Pre-conditions:

- Mechanisms to enforce that encryption keys are only accessible by properly authorised object store
- Mechanisms that enforce that the object store only makes use of correct and properly authenticated encryption keys.
- Mechanisms to enable client to turn on encryption for their outsourced data

- Post-conditions:

- Client is able to leverage an (off-premise) encryption-enabled object storage to store its data

3.3.1.3 Client retrieves data on encrypted object storage

- Actors: Clients and the object storage infrastructure, where client data reside in encrypted form

- Purpose: Clients are able to correctly retrieve their data, whose storage is outsourced to an encryption-enabled object store

- Priority:

- On the object storage side:

MUST: Data decryption capability assuming access to a key manager

COULD: Secure and scalable key management

COULD: Stored data integrity protection/check

- On the client side:

COULD: Secure key management

- Pre-conditions:
 - Mechanisms to enforce that encryption keys are only accessible by a properly authorised object store.
 - Mechanisms that enforce that the object store only makes use of correct and properly authenticated encryption keys.
- Post-conditions:
 - Client can effectively retrieve data stored in an encryption-enabled object store.

3.3.1.4 Client configures policies for secure deletion of their data

- Actors: Clients and the (private or public) object storage infrastructure, where client-data is stored.
- Purpose: Clients are able to configure secure deletion policies for their object store data.
- Priority:
 - MUST: Mechanisms that enable the client to configure data deletion policies.

- Pre-conditions:
 - Proper client authentication on the storage backend.
 - If the object store is off-premise, infrastructure for client-specific accounts on the storage backend, where policy configuration can take place.
- Post-conditions:
 - Client are able to configure policies for the secure deletion of their data that reside on an object store backend.

3.3.1.5 Client securely deletes data on encrypted object storage

- Actors: Clients and an object store, where client-data is stored.
- Purpose: Data that reside on the object store should be possible to be securely deleted upon client request. That is, securely deleted client data should be irrecoverable even by an adversary who has access to the entire object store infrastructure.
- Priorities:
 - MUST: Support for secure deletion, when key management is performed outside the premise of object store infrastructure.

- COULD: Support secure deletion, assuming an attacker who has access to the key management premise after secure deletion of data has been ordered. This requires support for secure deletion of keys, assuming that keys are stored within the premise of the object store, e.g., to a location that is easily controlled and designated to keys.
- Pre-conditions:
 - Encryption-enabled object storage
 - Secure deletion enabled (secure) key management
- Post-conditions:
 - Client is enabled to configure secure deletion policies for their object-store data.

3.3.1.6 Client uses policies to securely delete data on encrypted object storage, i.e., securely delete data after certain conditions have been satisfied

- Actors: Clients and an object store, where client data is stored
- Purpose: Data that reside on the object store should be possible to be securely deleted upon certain client-defined (and object-specific) policies have been satisfied. That is, upon such policy is satisfied, the associated object(s) should be securely deleted, i.e., should be transition to a state that constitutes it irrecoverable even by an adversary who has access to the entire object store infrastructure.
- Priorities:
 - MUST: Support for policy-based secure deletion, when key management is performed outside the premise of object store infrastructure
 - COULD: Support secure deletion, assuming an attacker who has access to the key management premise after a policy associated to the secure deletion of one or more objects is satisfied. This requires support for secure deletion of keys, assuming that keys are stored within the premise of the object store, e.g., to a location that is easily controlled and designated to keys
- Pre-conditions:
 - Secure deletion policy specification and evaluation mechanism on a per-object basis
 - Encryption-enabled object storage
 - Secure deletion enabled (secure) key management
- Post-conditions:
 - Data is securely deleted, after the data deletion policy data is associated with is satisfied

3.3.1.7 Client makes use of scalable and secure key management service offered by object storage for his (private) object storage

- Actors: Clients who make use of an object store, and external key management service for object stores
- Purpose: Clients outsource the storage and management of their keys to an external key management service
- Priority:

- **MUST:** Secure key management on the service side: all the operations of key lifecycle have to be supported in a way that key or data availability is guaranteed; in addition keys should only be given access to properly authorized (and authenticated) entities, to ensure key confidentiality against unauthorized parties.
- **COULD:** Secure deletion of old keys
- **Pre-conditions:**
 - Capability of securely deleting objects
- **Post-conditions**
 - Secure key-management service is offered by an object store to clients.

3.4 Requirements for relevant Cloud Data Security Solutions

3.4.1 Introduction

3.4.1.1 Purpose

The encryption and key-management functions in the OpenStack platform and specifically in OpenStack Swift shall support encryption and secure deletion features. ESCUDO-CLOUD primarily contributes to these features through schemes for key management. In particular, secure deletion is realized by applying specific protection mechanisms based on attributes relating when data should be deleted securely. Later on keys relating to those attributes must be deleted. The mechanisms produced by ESCUDO-CLOUD should therefore support these goals in a flexible way.

3.4.1.2 Conventions

None

3.4.1.3 Major relevant ESCUDO-CLOUD dimensions

ESCUDO-CLOUD considers four dimensions. We highlight in which way they are relevant for the use case.

1) Security properties, including confidentiality, integrity, and availability

Confidentiality is the only dimension among those required for the key material used to encrypt data, and it is a key requirement. The other elements of the dimension are of little relevance.

2) Sharing requirements, such as access by the client and sharing with others

Since platform-hosted encryption solutions do not per se facilitate access by multiple clients (tenants) to the same data, and since the approach should also be provided in a way that is transparent to the client, this dimension is of low relevance.

3) Access requirements, including upload/download, fine-grained retrieval, and partial write operations

A large emphasis will be put on fine-grained control and attribute-based techniques, which allow protecting of data and file objects with mechanisms specific to attributes attached to the data. In particular, key management will use hierarchies of raw keys that enable encryption policies.

4) Cloud architectures, whether single-cloud or multi-cloud or federate clouds

The use case for encryption in an OpenStack storage platform targets only a single-cloud model, which works under the assumption of one centralized administration entity. Multiple clients (tenants) can access this cloud platform.

3.4.2 Overall Description

3.4.2.1 Solution Perspective

The architecture of a server-side cloud-storage encryption solution looks as shown in Figure 7. Within this architecture, multiple different clients may access the cloud service through interface nodes, which are usually stateless. These nodes perform data encryption and further protection operations. Extending this picture, the different key-management options attach to the interface nodes directly, some of them may be exploiting a security context established directly between a client and the key-management system. The solution provides cryptographic protection against an adversary that has access to the back-end of the system, for example, to the storage nodes or to the network connecting the interface nodes to the back-end.

3.4.2.2 Solution Functions

- Encryption for data-at-rest
- Access relevant encryption keys depending on client
- Provision tenant-specific key material and metadata for on-boarding and off-boarding of tenants
- Attach protection attributes to stored data, representing different access policies and deletion policies
- Destroy keys through the key manager and re-encrypt data that should remain accessible with fresh keys

3.4.2.3 Actors and Characteristics

Server operator - may access the storage nodes, internal network and more, but no encryption functions nor cryptographic keys.

Security operator - manages keys on behalf of the cloud-service operator or on behalf of clients.

Client - accesses data stored in the cloud-storage platform, encryption and other features are transparent to the client.

3.4.2.4 Operating Environment

This use case is intended for close integration and deployment together with OpenStack, in particular with the object-storage platform of OpenStack Swift. Through interacting with the Swift

open-source development community, it is also envisaged that some solutions will be taken up into the upstream distribution. The operating environment may be a stand-alone cloud-object store or an integrated solution, which works together with many components of the OpenStack cloud platform, such as keystone for authentication and Barbican for key management. Typically such integrated platforms are or will be offered by public clouds.

Technically OpenStack runs on Linux hosts and is realized in Python.

3.4.2.5 Design and Implementation Constraints

APIs and protocols have to be chosen in accordance with the OpenStack project.

Platform, language, and development tools must be the same as in OpenStack.

3.4.2.6 Assumptions and Dependencies

It lies beyond the control of ESCUDO-CLOUD whether functions developed in the scope of the project will be taken up into the OpenStack distribution by the open-source community. This is a process that takes time and considerable work. Due to the open-source nature of the solution, it can be developed and integrated as well without support from the community, however.

3.4.3 External Interface Requirements

3.4.3.1 User Interfaces

Server operators need access to provisioning the system functions; this interface is based on existing mechanisms, such as configuration files and command-line utilities.

Security operators will access the key-management systems. These interfaces will initially be based on command-line tools and may use a graphical user interface in a future version.

Clients access the cloud-storage system through REST calls and have no specific user interface for storage and retrieval functions. Depending on the integration of the OpenStack basic components with an existing client-facing user interface (such as the Horizon dashboard), some functions may be made available also in such user interfaces.

3.4.3.2 Hardware Interfaces

No specific hardware is needed.

3.4.3.3 Software Interfaces

Various libraries will be accessed by the implementation, for example, providing support for encryption and communication functions.

3.4.3.4 Communications Interfaces

Most functions of the service will be integrated with the ordinary OpenStack storage infrastructure, which uses a REST API standardized by the OpenStack framework. These will be extended for specific functions relating to the encryption and secure deletion capabilities.

A specific, new API may be introduced for supporting the key-management tasks.

3.4.4 System Features

3.4.4.1 Managing Infrastructure Keys for the Cloud Provider (IKM)

3.4.4.1.1 Description and Priority

This infrastructure-key management system takes care of keys used by the cloud platform and is operated by the cloud provider. It is usually transparent to the tenants and to the clients.

3.4.4.1.2 Stimulus/Response Sequences

An operator acting on behalf of the cloud provider can manage and perform all functions necessary for cryptographic processes in the infrastructure. This includes data encryption and possibly further functions, such as server authentication, data integrity, and secure deletion. The operations support the complete lifecycle of cryptographic keys and related cryptographic materials (create, read, update, and delete (CRUD) operations).

3.4.4.1.3 Functional Requirements

REQ-UC1-IKM-1: The system supports CRUD operations for cryptographic keys and related cryptographic material which is used in the cloud infrastructure.

REQ-UC1-IKM-2: Deployment and management of infrastructure keys is driven by policies and automated, to the extent possible.

REQ-UC1-IKM-3: The cloud infrastructure-key management system supports the relevant open standards that are used by the industry (OASIS KMIP for REST APIs and possibly OASIS PKCS #11 for interfaces to secure hardware modules).

REQ-UC1-IKM-4: The cloud infrastructure-key management system supports the secure deletion of cryptographic material.

3.4.4.1.4 Relevant ESCUDO-CLOUD project dimensions

T2.2 / WP2 and T3.2/WP3 support techniques for infrastructure-key management, focusing on secure deletion and multi-user access to cryptographic material. The key management concepts and solutions provided by T2.2 and T3.2 directly address the needs for infrastructure-key management.

3.4.4.2 Managing Tenant Keys in the Cloud Platform (TKM)

3.4.4.2.1 Description and Priority

This tenant-specific key-management system (TKM) takes care of keys that are under the ultimate control of the cloud tenants. Its keys and cryptographic material are governed by the tenants. For tight integration with the cloud platform, the TKM itself is usually operated by the cloud provider. The tenants and possibly also the clients may use the TKM for provisioning and managing cryptographic material that operates under partial control of the cloud-platform provider.

3.4.4.2.2 Stimulus/Response Sequences

An operator acting on behalf of a tenant can manage and perform all functions necessary for cryptographic processes, controlled by the tenant but performed by the infrastructure. This includes data encryption and possibly further functions, such as service endpoint authentication, data integrity, and secure deletion. The operations support the complete lifecycle of cryptographic keys and related cryptographic materials (create, read, update, and delete (CRUD) operations).

3.4.4.2.3 Functional Requirements

REQ-UC1-TKM-1: The system supports CRUD operations for cryptographic keys and related cryptographic material which is used by a tenant.

REQ-UC1-TKM-2: Deployment and management of tenant keys is driven by policies and automated, to the extent possible.

REQ-UC1-TKM-3: The cloud tenant-key management system supports the relevant open standards that are used by the industry (OASIS KMIP for REST APIs and possibly OASIS PKCS #11 for interfaces to secure hardware modules).

REQ-UC1-TKM-4: The cloud tenant-key management system supports the secure deletion of cryptographic material.

3.4.4.2.4 Relevant ESCUDO-CLOUD project dimensions

T2.2 / WP2 and T3.2/WP3 support techniques for tenant key-management, focusing on secure deletion and multi-user access to cryptographic material. The key management concepts and solutions provided by T2.2 and T3.2 directly address the needs of tenant-key management.

3.4.4.3 Scalability of Key-Management Systems (SKM)

3.4.4.3.1 Description and Priority

There may be multiple key-management systems in the cloud platform, including infrastructure-key management (IKM) and tenant-key management (TKM). Due to the elasticity and the scalability of the cloud platform itself, an important feature of any key-management system lies in high throughput, scalability, and adaptation to cloud-specific weak consistency models. More precisely, since the typical cloud platform does not support strong transactional semantics such as atomicity,

but only weaker forms such as "eventual" consistency, the key-management system must be able to cope with the relaxed consistency notions and must not interfere with the existing features of the cloud platform. In particular, a key-management system should not introduce a single point of failure, should not limit throughput, and should not rely on strongly consistent or atomic operations on behalf of the cloud platform.

3.4.4.3.2 Stimulus/Response Sequences

The key-management systems, when used with the cloud platform, do not limit scalability or throughput of the infrastructure.

3.4.4.3.3 Functional Requirements

REQ-UC1-SKM-1: Key-management systems can be operated in a redundant and fault-tolerant way and do not introduce any single point of failure.

REQ-UC1-SKM-2: Key-management systems do not limit the scalability of the cloud platform. They must either be offered with large enough throughput or they must be scalable on their own.

REQ-UC1-SKM-3: Key-management systems can cope with the weak forms of consistency found in cloud platforms such as OpenStack. In particular, the key-management systems provide support for eventual consistency of the underlying operations in the cloud platform.

3.4.4.3.4 Relevant ESCUDO-CLOUD project dimensions

The methods produced by T2.2 / WP2 and T3.2/WP3 support scalability, high throughput and weak consistency for the key-management solutions used by the cloud platform.

3.4.5 System Features

3.4.5.1 Managing Infrastructure Keys for the Cloud Provider (IKM)

3.4.5.1.1 Description and Priority

This infrastructure-key management system takes care of keys used by the cloud platform and is operated by the cloud provider. It is usually transparent to the tenants and to the clients.

3.4.5.1.2 Stimulus/Response Sequences

An operator acting on behalf of the cloud provider can manage and perform all functions necessary for cryptographic processes in the infrastructure. This includes data encryption and possibly further functions, such as server authentication, data integrity, and secure deletion. The operations support the complete lifecycle of cryptographic keys and related cryptographic materials (create, read, update, and delete (CRUD) operations).

3.4.5.1.3 Functional Requirements

REQ-UC1-IKM-1: The system supports CRUD operations for cryptographic keys and related cryptographic material which is used in the cloud infrastructure.

REQ-UC1-IKM-2: Deployment and management of infrastructure keys is driven by policies and automated, to the extent possible.

REQ-UC1-IKM-3: The cloud infrastructure-key management system supports the relevant open standards that are used by the industry (OASIS KMIP for REST APIs and possibly OASIS PKCS #11 for interfaces to secure hardware modules).

REQ-UC1-IKM-4: The cloud infrastructure-key management system supports the secure deletion of cryptographic material.

3.4.5.1.4 Relevant ESCUDO-CLOUD project dimensions

T2.2 / WP2 and T3.2/WP3 support techniques for infrastructure-key management, focusing on secure deletion and multi-user access to cryptographic material. The key management concepts and solutions provided by T2.2 and T3.2 directly address the needs for infrastructure-key management.

3.4.5.2 Managing Tenant Keys in the Cloud Platform (TKM)

3.4.5.2.1 Description and Priority

This tenant-specific key-management system (TKM) takes care of keys that are under the ultimate control of the cloud tenants. Its keys and cryptographic material are governed by the tenants. For tight integration with the cloud platform, the TKM itself is usually operated by the cloud provider. The tenants and possibly also the clients may use the TKM for provisioning and managing cryptographic material that operates under partial control of the cloud-platform provider.

3.4.5.2.2 Stimulus/Response Sequences

An operator acting on behalf of a tenant can manage and perform all functions necessary for cryptographic processes, controlled by the tenant but performed by the infrastructure. This includes data encryption and possibly further functions, such as service endpoint authentication, data integrity, and secure deletion. The operations support the complete lifecycle of cryptographic keys and related cryptographic materials (create, read, update, and delete (CRUD) operations).

3.4.5.2.3 Functional Requirements

REQ-UC1-TKM-1: The system supports CRUD operations for cryptographic keys and related cryptographic material which is used by a tenant.

REQ-UC1-TKM-2: Deployment and management of tenant keys is driven by policies and automated, to the extent possible.

REQ-UC1-TKM-3: The cloud tenant-key management system supports the relevant open standards that are used by the industry (OASIS KMIP for REST APIs and possibly OASIS PKCS #11 for interfaces to secure hardware modules).

REQ-UC1-TKM-4: The cloud tenant-key management system supports the secure deletion of cryptographic material.

3.4.5.2.4 Relevant ESCUDO-CLOUD project dimensions

T2.2 / WP2 and T3.2/WP3 support techniques for tenant key-management, focusing on secure deletion and multi-user access to cryptographic material. The key management concepts and solutions provided by T2.2 and T3.2 directly address the needs of tenant-key management.

3.4.5.3 Scalability of Key-Management Systems (SKM)

3.4.5.3.1 Description and Priority

There may be multiple key-management systems in the cloud platform, including infrastructure-key management (IKM) and tenant-key management (TKM). Due to the elasticity and the scalability of the cloud platform itself, an important feature of any key-management system lies in high throughput, scalability, and adaptation to cloud-specific weak consistency models. More precisely, since the typical cloud platform does not support strong transactional semantics such as atomicity, but only weaker forms such as "eventual" consistency, the key-management systems must be able to cope with the relaxed consistency notions and must not interfere with the existing features of the cloud platform. In particular, a key-management system should not introduce a single point of failure, should not limit throughput, and should not rely on strongly consistent or atomic operations on behalf of the cloud platform.

3.4.5.3.2 Stimulus/Response Sequences

The key-management systems, when used with the cloud platform, do not limit scalability or throughput of the infrastructure.

3.4.5.3.3 Functional Requirements

REQ-UC1-SKM-1: Key-management systems can be operated in a redundant and fault-tolerant way and do not introduce any single point of failure.

REQ-UC1-SKM-2: Key-management systems do not limit the scalability of the cloud platform. They must either be offered with large enough throughput or they must be scalable on their own.

REQ-UC1-SKM-3: Key-management systems can cope with the weak forms of consistency found in cloud platforms such as OpenStack. In particular, the key-management systems provide support for eventual consistency of the underlying operations in the cloud platform.

3.4.5.3.4 Relevant ESCUDO-CLOUD project dimensions

The methods produced by T2.2 / WP2 and T3.2 / WP3 support scalability, high throughput and weak consistency for the key-management solutions used by the cloud platform. This relates to the following ESCUDO dimensions identified before:

- Security properties: i) Confidentiality iii) Availability

The key-management solution primarily supports confidentiality, and its proper working is also needed to let clients access their data. In this sense, also availability is a relevant dimension.

- Access requirements: i) Upload/download; ii) Fine-grained retrieval

The key-management and encryption mechanisms shall support access as well as fine-grained and policy-based control over the protected data. Write operations are considered to be part of the “Upload/download” dimension.

- Sharing requirements i) Access by data owners; ii) Selective sharing with other users/owners

For the use case, no particular sharing requirements in terms of access by multiple different clients is foreseen. The primary focus lies on protecting data owned by a single client. Certain tasks will also focus on sharing data through the cloud with other clients, for which consistency is a concern.

- Cloud architectures: i) Single cloud provider

The key-management solution addresses a single cloud provider.

3.4.6 Requirements Catalogue

Table 2. Requirements Catalogue for Use Case 1

Requirement Reference #	Requirement Description	ESCUDO-CLOUD dimension	Priority	Dependencies on other Requirement	Relevant Core Work Package
REQ-UC1-IKM-1	CRUD operations for infrastructure keys	- Confidentiality - Availability	Medium		WP2 (T2.1, T2.2)
REQ-UC1-IKM-2	Policy-driven and automated infrastructure-key management	- Confidentiality - Availability - Fine-grained retrieval	Medium	REQ-UC1-IKM-1	WP2 (T2.1, T2.2)
REQ-UC1-IKM-3	Support for standard APIs and protocols in infrastructure-key management	- Single cloud provider	Medium	REQ-UC1-IKM-1	WP2 (T2.2)
REQ-UC1-IKM4	Support for secure deletion of cryptographic material	- Confidentiality	Medium	REQ-UC1-IKM-2, REQ-UC1-IKM3	WP2 (T2.1)
REQ-UC1-TKM-1	CRUD operations for tenant keys	- Confidentiality - Availability	High		WP2 (T2.2)

REQ-UC1-TKM-2	Policy-driven and automated tenant-key management	- Confidentiality - Availability - Fine-grained retrieval	Medium	REQ-UC1-TKM-1	WP2 (T2.2)
REQ-UC1-TKM-3	Support for standard APIs and protocols in tenant-key management	- Single cloud provider	High	REQ-UC1-TKM-1	WP2 (T2.2)
REQ-UC1-TKM4	Support for secure deletion of cryptographic material	- Confidentiality	High	REQ-UC1-TKM-2, REQ-UC1-TKM-3	WP2 (T2.1)
REQ-UC1-SKM-1	Redundancy and fault-tolerance in key-management systems	- Availability	High	REQ-UC1-TKM-1, REQ-UC1-IKM1	WP2 (T2.2)
REQ-UC1-SKM-2	Scalable design of key-management system	- Confidentiality - Availability	Medium	REQ-UC1-TKM-1, REQ-UC1-IKM1	WP2 (T2.2)
REQ-UC1-SKM-3	Key-management solutions support weakly consistent operations in cloud platform	- Confidentiality - Availability - Selective sharing with other users/owners	Medium	REQ-UC1-TKM-1, REQ-UC1-IKM1, REQ-UC1-SKM-1	WP2 (T2.2) WP3 (T3.2)

3.4.7 Other Non-functional Requirements

3.4.7.1 Performance Requirements

Encryption solutions involve an overhead and extra work on the servers in the storage platform. This cannot be avoided. Hardware-supported encryption technology (such as the AES NI instructions on Intel processors) should be exploited wherever available.

3.4.7.2 Compliance Requirements

The cryptographic algorithms should conform to relevant approved national and international standards (for example, FIPS and NIST guidelines issued in the United States).

Standards specific to cloud-computing security should be respected whenever possible. For example, the ISO 27000 framework specifies ISO/IEC 27017 (Information technology – Security techniques – Code of practice for information security controls based on ISO/IEC 27002 for cloud services). The use case implementation should respect the recommendations made in there.

3.4.7.3 Software Quality Attributes

Software solutions to be made available open-source should comply with the development and quality standards in the relevant open-source communities.

3.4.7.4 Business Rules

- The clients are able to enable/disable the encryption of data.

3.4.7.5 Other Requirements

Legal rules related to export and import of cryptographic techniques and algorithms must be respected.

4 Elaboration of USE CASE 2 with Security Requirements

4.1 Introduction

The proposed use case 2 is a specialization of secure enterprise data management in the cloud. Particularly, we chose a case where data sharing is crucial to the business objective, since then the cloud offers clear advantages to on-premise solution. Furthermore, we chose a case which is of high relevance and actuality in the current business transformation to the digital economy and digitization of products and services. We chose maintenance in the aerospace supply chain.

4.2 High Level “Business” Use Case

4.2.1 Introductory Scenario

The aerospace industry is dominated by US (Boeing, Lockheed-Martin, Northrop Grumman) and European companies (EADS and BAE Systems) that participate in a dense network of relationships. A lot of actors (organizations) participating in this network collaborate and compete with other actors at the same time.

In this industry, a profound reorganisation of supply chain started since the 1980s, as changes in the product technology have modified the industry competitive factors and the leader firm’s role.

Maintenance is “the process of ensuring that a system continually performs its intended function at its designed-in level of reliability and safety”. To be more clear, the maintenance activities on aircraft are short enough to allow them to remain available for scheduled service.

In particular, there are two different types of maintenance:

- the scheduled maintenance, that is conducted at pre-set intervals in order to ensure the aircraft safety (this is a preventive form of maintenance);
- the unscheduled maintenance, which is carried out in case of breakdowns and requires a longer time than scheduled maintenance since it may entail extensive testing, adjusting and a replacement or overhaul of parts or subsystems.

On the contrary, overhaul represents the biggest and most labour-intensive maintenance event; it can only be performed by maintenance organizations able to satisfy special qualification

requirements. For example, to overhaul an aero engine it is necessary to remove, disassemble, clean, inspect, repair and test it using factory service manual approved procedures; when the engine is overhauled it will perform as new.

It is possible to put in evidence three different levels of MRO service providers:

- Original Equipment Manufacturers (OEM) ;
- Specialized third party contractors;
- Small enterprise for minor MRO activities.

The objectives of all MRO service providers can be summarized in the following points:

1. to ensure or restore safety and reliability of the equipment;
2. to have all products and processes information so that maintenance can be optimised when there aren't the right safety or reliability levels;
3. to have all information necessary to repair components or to design tooling if some items have to be repaired or replaced during the overhaul process;
4. to accomplish the previous objectives staying within time and cost budget.

4.2.2 Purpose

A central issue for maintenance and support service providers concerns the management of the growing amount of information generated by the development of highly complex aircraft systems and by stakeholders' requirements in terms of dependability increase and Life Support Cost (LSC) decrease. To face these problems, maintenance and support actors are depending more and more on ICT solutions. These are one of the main elements not only to improve the effectiveness and efficiency of the maintenance process for complex systems with a long lifecycle, but also to reduce the associated risks and to contribute to a more efficient business process. The benefits linked to the use of ICT systems in this business segment are:

- more controlled content sharing;
- information exchange and knowledge management;
- coordination of maintenance process with other processes;
- connection to strategic business objectives and external stakeholder requirements.

The technological challenge of this use case is to develop new supply chain cooperation systems, based on encrypted database technology. This technology is based on the search and aggregation of encrypted data and can be applied in planning the MRO service to different customers knowing neither the actual status of the aero fleet nor the capacity usage of the service provider, nor the inventory status.

4.2.3 Comparison to current practice

Traditionally, each party on each stage of the supply chain has its rather isolated forecasting processes which are mainly based on data of historical demand that aroused from their direct customers. The problem with these orders from the next stage is that they are again results of an isolated forecast and in general don't match the actual sales on the buyer's stage. Instead, they tend to have a larger variance. This effect of demand distortion results in amplified forecasting numbers

and dramatic swings in demand increasing with every step on the supply chain further away from the end customer. This phenomenon is known as the bullwhip effect.

Besides this rather technical problem, forecasters have to deal with the uncertainty given by a constantly changing market environment. New products, promotions or other first-time events can't be predicted by only using historical data. However, this information does exist, and CF is an attempt to bring them together to create a single, more accurate forecast which is accepted by all collaborating members of the supply chain. In a collaborative forecasting process ideally all supply chain members add their particular expertise and information to find the best possible forecast. The information about end customer demand is shared with the upstream supplier, so demand distortion can be reduced drastically. This again will drastically reduce the bullwhip effect.

Yet, despite the theoretical benefits of CF are well known, the practical implementations are rather scarce. A basic form of CF is that the buyer shares his forecast with the upstream suppliers, who can interpret this information as advanced notification of future orders and adapt their production plans. This is based on the assumption that the buyer has more accurate information about the actual end customer demand due to its position further downstream. However, in this constellation the buyer might try to influence the production plans of his supplier by manipulating the shared forecasts. If there are no methods applied to make this behaviour unattractive for the buyer, the supplier has to expect it and as a consequence would not take the forecast seriously. Companies like SUN, Hewlett-Packard and Texas Instruments implemented this kind of CF within so called quantity-flexible contracts. The forecasted quantities could be re-adjusted within certain boundaries, to increase their accuracy and therefore the acceptance by the supplier.

4.2.4 Objectives

In general, two main business-optimizing services have to be guaranteed in the aero engine overhaul supply chain: the collaborative demand forecasting, and the collaborative planning and scheduling of the overhaul activities. The first one allows MRO service provider to obtain demand forecasts from all customer based on on-condition engine status observations, reducing so overall costs due to a more accurate capacity planning; while the collaborative planning and scheduling guarantees better supply chain performance since an ideal receipt point for each engine can be computed (this is a way to decrease costs and TAT).

4.2.5 Stakeholders

In the aeronautic industry it is possible to analyse vertical and horizontal relationships among the different firms that contribute to the aircraft production.

Vertical relationships occur between the leader firm and the other firms taking part in the aeronautic program. This type of relationships underlines a complex and hierarchical organization at the base of aircraft production system.

The supply chain can be depicted as a pyramid where at the top there is a leader firm (as Boeing and Airbus) or a consortium that is responsible for the whole program and the assembly of the aircraft.

Furthermore, the leader firm organises the flow of the parts, components and systems; stores all products' relevant information in order to have the history of each component; manages relations with the final customer (airlines or aircraft leasing companies).

The first level is divided into three sub-sectors (airframe, equipment and avionics, and engines) with their own structure and a degree of autonomy associated with the details of the program. In this level there are those large firms (as General Electric Aviation, Rolls-Royce, Honeywell and Pratt & Whitney for the engine sector) that realise (or assemble) complex parts of the aircraft, such as the engine (or wings, tail, fuselage sections). These firms, on the base of all program specifications received from the leader, decide what they will produce in-house and what will be outsourced to third level suppliers. They can choose the second and third level suppliers, but they must deliver to the leader all parts and components realised with the related information so that it is easy to understand if the conditions of the contract and the accuracy of the production process are respected.

The second and third level is generally constituted by medium (and small) firms; their work is mainly coordinated and checked by their customer, and in many cases also by the program leader, in order to verify the quality standards and the production processes.

Horizontal relationships among firms that belong to the same pyramid level changed and developed over the last 50 years. In the 1950s, an aircraft was designed by one firm which was able to sustain both technological and economical efforts. In this way, only one firm was responsible for the program, and co-operation agreements didn't yet exist. Today, aerospace industry is completely different: this is characterised by collaborative and competitive relationships among firms, in order to spread, in an easier way, technologies and know-how within the industry. For example, Honeywell and Rolls-Royce are competitors, but they might collaborate and trade with each other too.

4.3 Use Cases

4.3.1 Use Case: Collaborative Forecasting in Aerospace Fleet Management

4.3.1.1 Actors

- Maintenance, Repair and Overhaul (MRO) service provider (usually a mid-size company) delivering aerospace parts
- Airlines running airplanes and outsourcing MRO

4.3.1.2 Purpose

Supply chain collaboration is the alignment of individual plans and strategies of the involved parties. The stronger coordination and the overcoming of information asymmetries with partners shall conduce to improvements of the supply chain performance. Additionally, the uncertainties in demand may be reduced by taking into account interactions at other levels of the supply chain.

Better knowledge of the downstream demand enables the customer to facilitate the vendor's predictability skills concerning his production and delivery capacities by providing him with appropriate data. Reduced inventory and hence lower costs are potential benefits for the partners.

4.3.1.3 Priority

Here, a general view on the main CPS functional requirements is provided:

1. Compute forecast on overhaul services demand.
Such functional requirement is linked to the first one: the monitoring of the engines working status. Thus, the knowledge of forecasts about the overhaul services demand allows MRO service provider to define its service plans in advance, avoiding periods of over and under capacity.
2. Plan overhaul activities for the next 3 (or 6) months.
The MRO service provider can have its overhaul activities planned: for example, it will be possible to know the best day in which an engine has to be shipped, or which human resources will be necessary to perform activities, as well as the time necessary to satisfy the customer demand.
3. Compute penalties for delays in the overhaul process.
The MRO service provider can have knowledge on the penalties related to possible delays in the customer demand satisfaction. Hence, it will be able to organize its service plan adopting a priority-based strategy.
4. Compute parts needs and propose purchasing orders.
The continuous monitoring of the service provider inventory (in terms of number of items in the warehouse) makes it possible to define which parts are needed (in function of customer demand) and to highlight the necessary purchasing orders.
5. Compute forecast on parts requests.
The CPS provides to parts suppliers forecasts on which and when parts will be required by a certain MRO service provider. Such information is important for suppliers since, in this way, they can estimate in advance the necessary resources to face the service provider demand.

4.3.1.4 Sequence of events

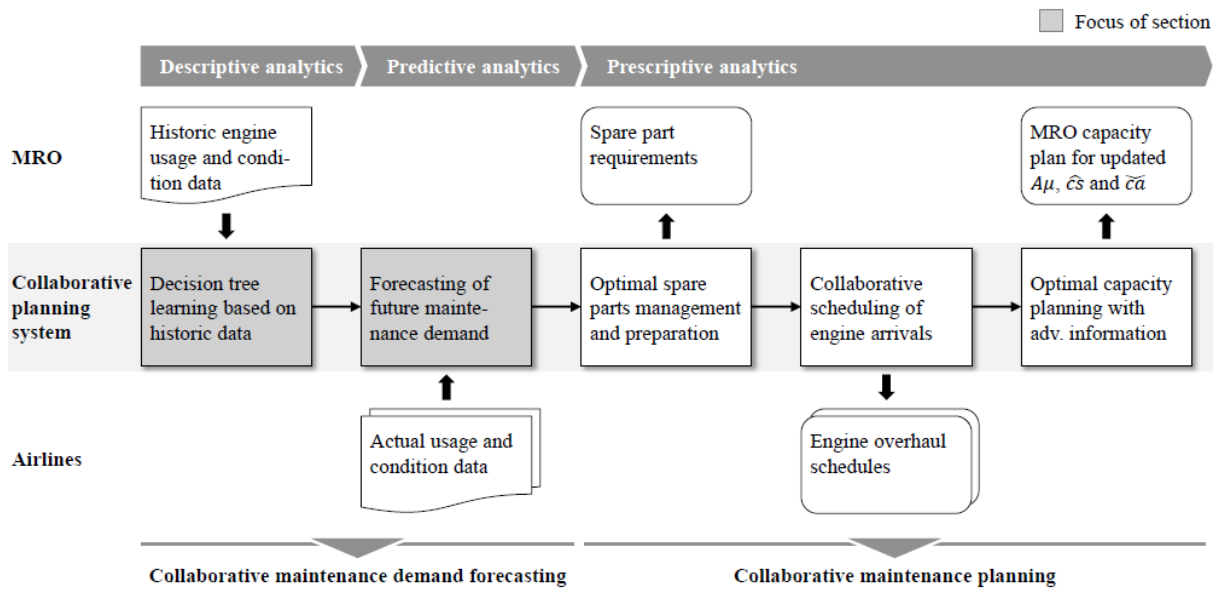


Figure 9. Sequence of events for Use Case 2

4.3.2 Non-functional Requirements of the use case

Information sharing within supply chains allows all involved actors to have better performance, but it is not costless: making information available to a common computing system requires investment in information technologies (ITs) and leads to risks of leaking confidential data.

In the following analysis, we consider risk related to the probability of confidential data leakage to supply chain partners or IT service provider.

In the case of the aero engine overhaul supply chain, risks for the appropriation of confidential data are different depending on the victim (the actor whose data are accessed) and the attacker (the actor who accesses private data). In the following, such risks are:

1. *Loss of information advantage.* Disclosing sales information to other organizations may change relationships among partners in a supply chain, since it may reduce the “information rents” effect for which especially a weak partner profit;
2. *Reconstruction of strategic decisions.* Sharing of data increases the visibility of operations for all companies involved; thus if confidentiality is not preserved, a competitor could anticipate company’s future plans;
3. *Development of a competitive product/service.* With the knowledge of suppliers, competitors can develop new competitive products or offer more competitive services;
4. *Weakening of the bargaining power after disclosure of purchase or supply volume.* A customer may compare its own purchase volume to that of other customers in order to calculate its share; such information can be used to strengthen its bargaining power. At the same way, suppliers can calculate their share of overall supply; this information increases their power over the customer in a business negotiation.

Airline company risks reducing its market share and profit with respect to competitors: if fleet status is visible to another airline, it can apply an aggressive market strategy targeted on the less efficient routes/aircrafts (*reconstruction of strategic decisions*). The risk run against the MRO service provider is related to the reduction of bargaining power since MRO service provider needs and profitability can be better evaluated (*weakening of the bargaining power after disclosure of purchase volume*). Lastly, the risk run by airline/air force against parts' suppliers is related to the probability that suppliers renegotiate the delivery deadlines according to their production needs (*weakening of the bargaining power after disclosure of purchase volume*), as well as they offer airline/air force information to outsiders (*loss of information advantage*).

At the same way, the MRO service provider could have its confidential data attacked from airlines, other MRO service providers or suppliers. In the first case, the service provider runs the risk to lose bargaining power during trading relationships with the airline/air force (*weakening of the bargaining power after disclosure of supply volume*). In fact, from information on resources availability, inventory management and Turn Around Time, effective service quality level can be inferred (and higher penalties can be imposed). In the second case, data of the MRO service provider are attacked by a direct competitor (also user of the cloud system); here the risk is a reduction of competitive power and so of market share (*loss of information advantage* and *development of a competitive product/service*). The competitor, in fact, can adopt a production and/or market strategy focused on stealing customers to the data owner. At last, in the case a supplier access service provider data, it can negotiate product deliveries deadlines threatening the manufacturing plan (*weakening of the bargaining power after disclosure of purchase volume*), or can sell data to other service providers' or to airlines (*loss of information advantage*). In this way, risks are related to the loss of control on manufacturing plans and service quality, and to the decrease of market share.

To conclude, suppliers risk a reduction of their market share against competing suppliers (*development of a competitive product/service*). Against MRO service providers, they risk losing its bargaining power and profits, i.e. due to higher penalties (*weakening of the bargaining power after disclosure of supply volume*); while they risk their private data are sold elsewhere if other participants (i.e. airlines/air forces) access them (*loss of information advantage*).

It is also important to consider that if data are known by an actor external to a specific supply chain (like a cloud service provider), the victim competitive position will be affected since the attacker could offer private data to another party in exchange of economic benefits (*loss of information advantage*). In general, greater the quantity of violated information higher is the risk that the victim reduces its competitive power.

4.4 Requirements for relevant Cloud Data Security Solutions

4.4.1 Introduction

4.4.1.1 Purpose

The planning and forecasting application for aerospace maintenance will be built on top of a secure (encrypted) cloud database. Additional components will be implemented as lightweight client-side application, e.g. written in JavaScript. Hence, the cloud security requirements stem from the cloud database. If the cloud database can fulfill the necessary security requirements for this business application so will the entire application and with it many more.

4.4.1.2 Major relevant ESCUDO-CLOUD dimensions

ESCUDO-CLOUD considers four different dimensions that help provide data owners with the combination of security and flexibility when trying to outsource their data resources to the cloud. These dimensions are shown in Table 1.

For the aerospace maintenance cloud application the following requirements are important and can be further detailed:

- Security properties

Confidentiality is of the utmost importance. Data from individual airlines and suppliers is not supposed to be uncontrolled shared with either the cloud provider or other users of the system. Access to this data would entail significant business risks. Integrity and availability can be partly dealt with existing cloud system means, but **integrity of access control decision** is very important again. An access decision should not be delegated to the cloud provider.

- Sharing requirements

Sharing requirements are at the core of use case 2. The data owners want to retain control over their data, but enable others, such as the MRO provider, to improve operations. As such data is supposed to be **shared in a selective manner with others**, policies need to be administered, including access restricted to the result of the data analysis.

- Access requirements

The aerospace maintenance use case is defined by **fine-grained access** as in a database system. Data is structured and processed in a common relational database system. **Data may be added**, but also sometimes **updated or deleted**.

- Cloud architectures

The use case is initially targeted for a **single cloud architecture** as found in most business application. Extensions to multi-cloud architecture are imaginable, but not in the initial focus. Furthermore solutions supporting only multi-cloud architectures are out of focus.

4.4.2 Initial solution architecture in the context of the use-cases

The solution will be based on an encrypted cloud database where the encryption key resides at the client. All database operations will be performed over encrypted data. The application programmer will continue to use the database using SQL. The database driver will encrypt query parameters and decrypt query results. The database engine will operate on encrypted data.

The challenge for supporting the aerospace maintenance use case is to support **fine-grained access control** requirements in addition to **encrypted query processing**. This necessitates encrypting data with different keys and database operators operating on data encrypted with different keys. The ESCUDO-CLOUD project will deliver such a solution.

4.4.3 Overall Description

4.4.3.1 Solution Perspective

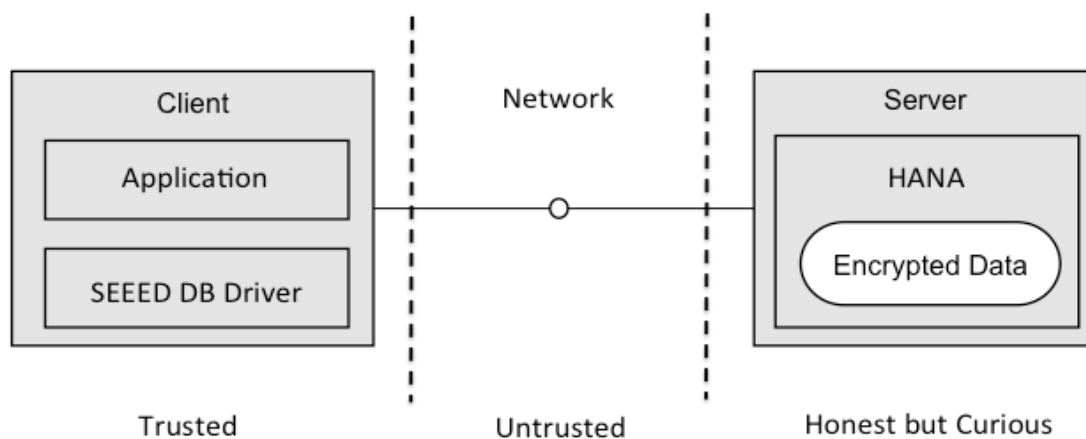


Figure 10. Trust model for the Use Case 2

The solution is based on the existing work SAP Security Research Karlsruhe in the SEED project also featured in the EU funded project PRACTICE and TREDISEC. The work extends the basic architecture as depicted in the Figure 10 above by multiple clients accessing the database with multiple keys. This has significant implications on the architecture of the database operators.

4.4.3.2 Solution Functions

We structure the solution functional requirements including security functions as follows:

ESCUDO-CLOUD Dimension	Function	Detailed Requirement
Access Control	Access control per client	
	Access control per group of clients	
	Access control per database cell	
	Access control matrix model	
	Access grant and revoke by administrator	
	Access control enforced by client	

Key Management		One key per client	
		Group key management	
		Client key securely stored at client only	
		Group keys derivable	
Encrypted Query Processing		Encryption schemes	Deterministic
			Order-Preserving
			Probabilistic
			Additively homomorphic
		Adjustable onion encryption	
		Query processing by	Proxy re-encryption
			Query rewriting
			Post-processing in the database driver
		Support for different keys	

4.4.3.3 Actors and Characteristics

- Clients: Airlines, Suppliers, MRO provider

These actors intend to use the business functionality of the application. They expect their security requirements driven by business and operational needs to be met.

- Cloud Service Provider

The CSP is a typical SaaS provider that manages the database and the use case application on the behalf of the clients.

4.4.3.4 Operating Environment

The application and the database will be based on SAP HANA and the SAP HANA Cloud Platform.

4.4.4 External Interface Requirements

4.4.4.1 User Interfaces

The application will have a web-based interface.

4.4.4.2 Hardware Interfaces

The cloud service provider will be bound to hardware supported by the SAP HANA database, i.e. the Intel XEON platform.

4.4.4.3 Software Interfaces

We will expose a JDBC programming interface supporting standard SQL.

4.4.5 System Features

4.4.5.1 Access control

4.4.5.1.1 Description and Priority

Each client is supposed to define and enforce access rights for the data owned by him. Sharing is enabled by granting access to other partners.

4.4.5.1.2 Functional Requirements

REQ-UC2-AC1: Access control decisions should be based on the subject of a client, i.e. the subject granularity is a client. It is at the discretion of a client whether he intends to enforce finer-granular access control, e.g. at the level of the user. This can be either achieved by increasing the number of clients (at the corresponding performance penalty) or enforcing regular access control at the database driver.

REQ-UC2-AC2: It should be possible to group several users into a group and grant or revoke access for the entire group.

REQ-UC2-AC3: Access control decisions should be based on the object of a database cell as identified by a column (in a table) and a row owner. It should be possible to group several cells (in the same column) for the same access rights.

REQ-UC2-AC4: The access control model should be an access control matrix. Advanced models, such as role-based access control or attribute-based access control, are initially out-of-scope. These advanced models can be mapped into the access control matrix using additional tools, but these are also initially out of scope.

REQ-UC2-AC5: Access control rights should be grantable and revocable by the database administrator with support of the data owning clients. A grant or revoke operation is triggered by the database administrator, but the necessary adaptations, such as the revelation of a key or the re-encryption of data must be performed by a client. Revoking operations are supposed to be infrequent.

REQ-UC2-AC6: Access control should be enforced by the client, i.e. cryptographically enforced. Access control should not be entrusted to the cloud service provider. The integrity of access control decision is of the utmost importance to the clients. The clients should stay in control of their data.

4.4.5.1.3 Relevant ESCUDO-CLOUD project dimensions

- Security properties

Access control ensures **confidentiality** to the cloud provider or other users of the system. **Integrity of access control decisions** is dealt by cryptographically enforced access control.

- Sharing requirements

Managing access control policies enables **selective sharing with others**. The subject of access control can be either clients or groups of clients, enabling a hierarchical model.

- Access requirements

The relational database model requires **fine-grained access** on a database cell level. **Data may be added**, but also **updated or deleted**.

4.4.5.2 Key Management

4.4.5.2.1 Description and Priority

Each client is supposed to retain control of the encryption key of its data. Keys may be shared for selective access by others. The cloud service provider is not supposed to learn any keys.

4.4.5.2.2 Functional Requirements

REQ-UC2-KM1: Each client should have its own key generated and kept confidential at its site. This key is not supposed to be shared. From its master key other keys for access to finer-granular objects (such as a database column) may be derived.

REQ-UC2-KM2: There should be keys for access by groups. These keys are shared by groups of clients to access common data. From these keys other keys for access to finer-granular objects (such as a database column) may be derived.

REQ-UC2-KM3: Client keys should be stored securely, e.g. in a secure key store (PKCS#12) protected by a password. Additional mechanisms such as hardware security modules are initially optional.

REQ-UC2-KM4: Groups key may be derived using a public key hierarchy stored at the cloud service provider. This is a low priority feature.

4.4.5.2.3 Relevant ESCUDO-CLOUD project dimensions

- Security properties

Key management supports **confidentiality** to mainly the cloud provider who is processing queries on encrypted data. Key management enables **integrity of access control decisions**.

- Sharing requirements

Group keys enable **selective sharing with others** by selectively sharing keys. Group key hierarchies can simplify administration

4.4.5.3 Encrypted Query Processing

4.4.5.3.1 Description and Priority

The cloud service provider is supposed to perform the processing of database queries on encrypted data. This ensures the confidentiality of the owners' data and enables control of the clients on access control decisions.

4.4.5.3.2 Functional Requirements

REQ-UC2-EQ1: The database should support different encryption schemes for different database operations. Mostly notably, the database should support order-preserving encryption for range and rank queries, deterministic encryption for equality selection/joins, and grouping, probabilistic encryption for retrieval and count operations and additively homomorphic encryption for summation. The additively homomorphic database operator requires a change in implementation, since addition needs to be replaced by modular multiplication. This needs to be supported. Other database operations may be adapted simply by adapting the type.

REQ-UC2-EQ2: The encryption should be adjustable to the database operations performed. Probabilistic encryption is more secure than deterministic encryption which is more secure than order-preserving encryption. Yet, order-preserving encryption enables strictly more database operations than deterministic encryption which enables strictly more database operations than probabilistic encryption. Hence, we wrap an order-preserving encryption in a deterministic encryption wrapped in a probabilistic encryption. Depending on the functionality needed by a query we decrypt to appropriate layer. Order-preserving encryption will always be maintained.

REQ-UC2-EQ3: In order to adjust the encryption and operate on different keys we need to analyze and evaluate the query differently. The database driver should at least support three different query evaluation techniques: query rewriting, proxy re-encryption and post-processing. Query rewriting should be able to encrypt parameters under the different keys and split the queries over multiple tables, one for each access group. The result should be combinable. Proxy re-encryption should be able to adjust the keys of different columns for comparison, e.g. in an equality join across different access groups. This may entail a new proxy re-encryption scheme. Post-processing should support combining query result of different keys, e.g. aggregations under different keys in additively homomorphic encryption. Post-processing should also enable processing sequential operations with incompatible encryption schemes, such as sorting a homomorphic encryption.

REQ-UC2-EQ4: All database operations should be supported across different client keys, i.e. spanning multiple access groups.

4.4.5.3.3 Relevant ESCUDO-CLOUD project dimensions

- Security properties

Encrypted query processing ensures **confidentiality** to the cloud provider.

- Sharing requirements

The support of different keys per client and access group enables **selective sharing with others**.

- Access requirements

The relational database model enables **fine-grained access** including processing as implied by a query language as SQL. **Data may be added**, but also **updated or deleted**.

4.4.6 Requirements Catalogue

Table 3. Requirements Catalogue for Use Case 2

Requirement Reference #	Requirement Description	ESCUDO-CLOUD dimension	Priority	Dependencies on other Requirement	Relevant Core Work Package
REQ-UC2-AC1	Access control per client	- Confidentiality - Selective sharing with other users/owners	High	REQ-UC2-KM1	WP2
REQ-UC2-AC2	Access control per group of clients	- Confidentiality - Selective sharing with other users/owners	High	REQ-UC2-KM2	WP3
REQ-UC2-AC3	Access control per database cell	- Confidentiality - Fine-grained retrieval	High	REQ-UC2-EQ2	WP3
REQ-UC2-AC4	Access control matrix model	- Confidentiality - Access by data owners - Selective sharing with other users/owners - Fine-grained retrieval	Medium		WP2/WP3
REQ-UC2-AC5	Access grant and revoke by administrator	- Confidentiality - Integrity - Selective sharing with other users/owners	Low	REQ-UC2-EQ3	WP2/WP3
REQ-UC2-AC6	Access control enforced by client	- Confidentiality - Integrity	High	REQ-UC2-KM3	WP2/3

REQ-UC2-KM1	One key per client	<ul style="list-style-type: none"> - Confidentiality - Integrity - Access by data owners 	High		WP2
REQ-UC2-KM2	Group key management	<ul style="list-style-type: none"> - Confidentiality - Integrity - Selective sharing with other users/owners 	High		WP2/WP3
REQ-UC2-KM3	Client key securely stored at client only	<ul style="list-style-type: none"> - Confidentiality - Integrity - Access by data owners 	High		WP2
REQ-UC2-KM4	Group keys derivable	<ul style="list-style-type: none"> - Confidentiality - Integrity - Selective sharing with other users/owners 	High	REQ-UC2-KM2	WP2/WP3
REQ-UC2-EQ1	Encryption schemes	<ul style="list-style-type: none"> - Confidentiality - Fine-grained retrieval - Write operations 	High		WP3
REQ-UC2-EQ2	Adjustable onion encryption	<ul style="list-style-type: none"> - Confidentiality - Fine-grained retrieval - Write operations 	High	REQ-UC2-EQ1	WP3
REQ-UC2-EQ3	Proxy re-encryption, Query rewriting, Post-processing	<ul style="list-style-type: none"> - Confidentiality - Fine-grained retrieval - Write operations 	High		WP3
REQ-UC2-EQ4	Support for different keys	<ul style="list-style-type: none"> - Confidentiality - Fine-grained retrieval - Write operations 	High	REQ-UC2-EQ3	WP3

4.4.7 Other Non-functional Requirements

4.4.7.1 Performance Requirements

The performance of database queries under reasonable (but small) sharing requirements should remain acceptable from a user perspective.

5 Elaboration of USE CASE 3 with Security Requirements

5.1 Introduction

The main objective of this business use case is to allow a BT customer to protect and control their confidential and sensitive information with a user-friendly data protection service that will keep data private and help meet regulatory compliance requirements. Customers will be able to store their data on multiple cloud vendors and platforms, and will be able to manage the security related aspects of their stored data via the federated protection service. Only the customer (or a trusted third party) will have the access and control of the decryption keys, giving the freedom to decrypt data on-demand and in real time.

Problem statement: A typical BT customer wants to outsource a large amount of its data assets to a scalable, reliable, and fault-tolerant storage service, and retain data protection control to meet privacy and regulatory compliance requirements.

The problem to be solved is the security and management of data that is hosted on third party infrastructures, for example in the form of block storage, data backups, or an application database. The problem is further complicated in the cloud computing environment as the data can be replicated and moved automatically to cater for the scalability and reliability needs of the cloud providers customers, thus increasing the risk of a security compromise. In addition to the data security concerns, most customers also have to abide by their company's data protection policies and governmental regulatory compliance (e.g., HIPAA, HITECH, Sarbanes-Oxley, GLB, and PCI DSS). Furthermore, a customer can make use of many other cloud service providers offering storage services, in addition to BT Cloud Compute, with each cloud provider offering its own Application Programming Interface (API), specialised services, and security functionalities to satisfy various user requirements. Therefore, in this use case we will provide a cloud security service that provides data protection service to a customer for block storage and object storage that works seamlessly across multiple cloud vendors. This will be initially validated on BT's Cloud Incubator and exploited via BT Cloud Compute.

The federated secure cloud storage use case will be able to utilise the resources and capabilities offered by ESCUDO-CLOUD to protect customers' data in a cloud environment. For instance, the fine-grained access control features provided by ESCUDO-CLOUD can be used to authenticate entities (e.g., customers, virtual machines) to grant them the permission to call and use APIs of the different cloud platforms being used in this use case. This use case will also aim to leverage the key

management solution provided by ESCUDO-CLOUD to ensure that only the authorised entities can be provisioned with keys and are able to decrypt block devices and objects. Lastly, the availability of ESCUDO-CLOUD multiple cloud architectures will help in the integration efforts needed to federate the secure storage across the multiple cloud service providers.

5.2 High Level “Business” Use Case

5.2.1 Introductory Scenario

One of the basic requirements for almost all software applications is the ability to store and retrieve data, and to this end, on-demand and cost-effective storage is perhaps the most common service offered by any Infrastructure as a Service (IaaS) cloud service provider. This service is built on top of many physically and logically distributed storage resources located in the premises of the cloud service provider, but is exposed to the end-users and customers as a single and federated cloud storage service. BT Cloud Compute is responsible for providing high fault-tolerance and scalability for customers’ storage requirements, which it does through the use of techniques like redundancy, and replication and distribution of data near the points-of-demands.

BT Cloud Compute offers three kinds of storage services; block storage, object storage and Big Data storage services. The block storage service offers the creation and management of virtualised raw block devices of a user-specified size. These block devices are typically used as hard disks by attaching them to customers’ virtual machines using the cloud service’s API. The object storage service offers the storage and management of data as objects, instead of files and blocks. An object is typically comprised of the data, its metadata, and a globally unique identifier. The Big Data storage services are offered in form of Hadoop File System (HDFS) clusters.

In order to deploy and manage the resource provisioning workflows for the softwares and services necessary for this use case, we utilise a cloud service store. In the context of ESCUDO-CLOUD, the Data Owner can trust the service store and the data protection service being offered through it whereas it places limited trust on the cloud service provider(s) that will eventually host the stored data. Therefore the data protection solution has to cater for the Honest-but-Curious model, in which the cloud service providers store the outsourced data without tampering it, and retrieve the data on request from the owner without any malicious modifications. However, the cloud service providers may try to look in to or analyse the stored data.

Hence, the service store offers an independent access control as a service that enforces the data-at-rest encryption of the data stored in the infrastructure of multiple cloud platforms. An integral part of the access control as a service feature is the management of the keys used to encrypt the stored data. Therefore, key management as a service feature also has to be a part of the service store offerings but in this case it is tightly coupled with the access control service. This tight coupling is required because in this use case only the data owners are responsible for the enforcement of data encryption, and they do this through a policy based framework that is able to integrate the access control rules and mechanisms with the encryption keys that can be used to access or retrieve the

data. This fits perfectly with one of the main goals of ESCUDO-CLOUD, that is, to give the control of data to the data owners. An overview of the overall BT use case is shown in Figure 11.

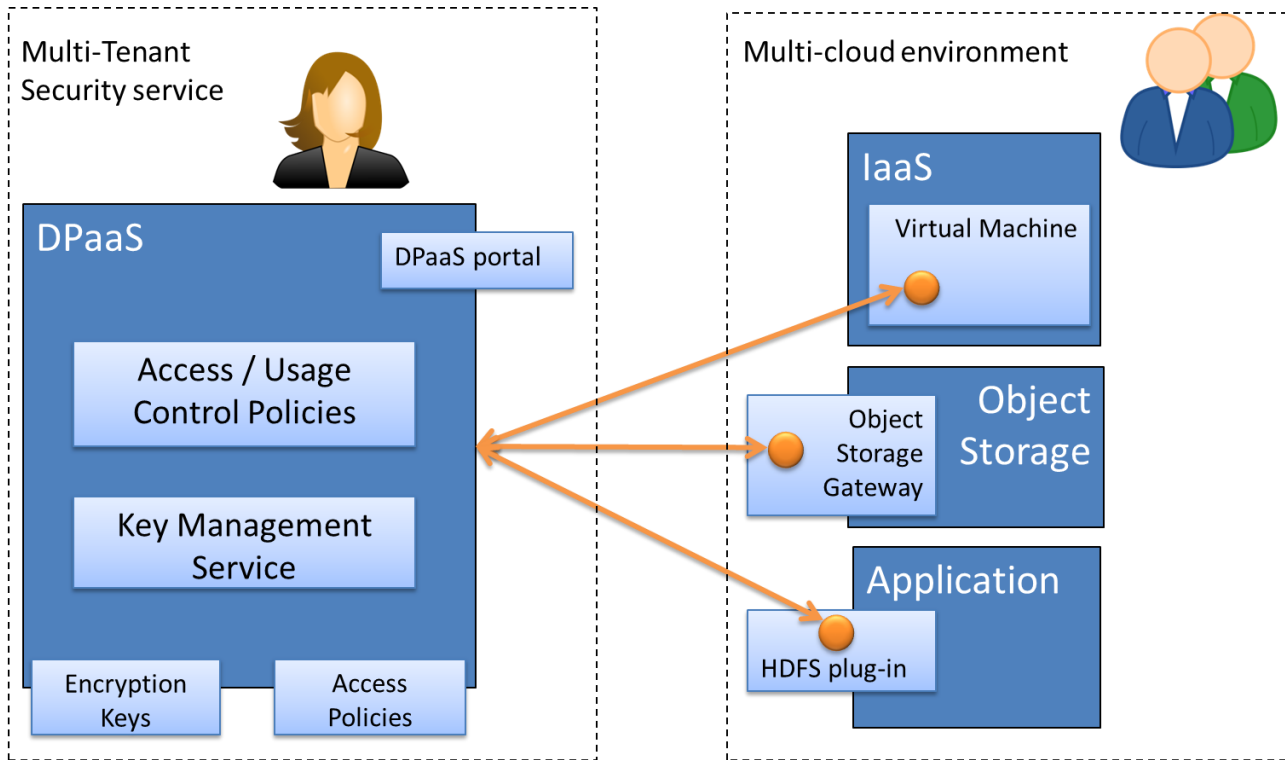


Figure 11. High level view of the Data Protection as a Service use case

Therefore, the data protection is enforced through the tight coupling of the access control service and the key management service, and the abstraction of the various storage services offered by multiple cloud platforms via the cloud service store.

Characterisation: This use case requires explicit consideration of a multi cloud environment and the consideration of data protection offered as a service in that environment. The use case is driven by BT, which enables its immediate deployment and exploitation in the BT Cloud Compute platform. EMC will provide support for the definition of requirements from regulations, and also seeking exploitation of the solutions. The key features of the data protection as a service model are:-

- An independent service offered via the cloud service store that enables customers to protect data stored on multiple cloud platforms
- The data is protected by encrypting it on the client-side, and ensuring that cloud service providers have no access to the encryption keys or protection policies.
- The encrypted data can be stored on multiple cloud storage services like virtual volumes, object stores and big-data clusters etc.
- Access control and key management are offered as independent but tightly coupled services that manage the protection of the data via an integrated policy framework.
- Decryption of the protected data is only possible following a policy-based approval procedure and the resulting release of the encryption key.

- The encryption and decryption process is transparent to applications and end-users while data stays always encrypted in the cloud.
- It integrates seamlessly with cloud service stores and works with most cloud platforms.
- It complies with specific data security regulations or pre-configured compliance policies.

5.2.2 Purpose

The purpose of this use case is to demonstrate the capability of a BT cloud customer to provision and utilise, as needed, secure data storage across multiple cloud service providers. The data owners have the ability to outsource their data while maintaining control over them, with the ability to regulate access to them and share them with other users in a selective way. The data owners also have the assurance that their data will remain protected from the untrusted cloud service providers.

From a commercial point of view this provides value to the customer in terms of providing redundancy if they choose multiple service providers, resilience by having data stored securely remotely, market competition for cloud storage services – in particular through avoidance of lock in. This latter benefit should translate into cost savings. In addition, by retaining control of key management the customer has more flexibility in meeting regulatory and privacy requirements and ensuring data confidentiality and secure access.

5.2.3 Comparison to current practice

Today, users storing data in the cloud need to put complete trust in the cloud service providers that they will correctly store and manage the outsourced information. Although all of the cloud service providers claim to apply security measures in the storage services they are offering, these measures give full trust to the cloud service provider and thus allows it to have full access to the data. Representative examples of this type of storage providers are services like Google Docs [1], Dropbox [2] or Salesforce. They are able to offer advanced functions on the outsourced data, like extensive sharing and processing abilities, but only because they are able to access the plaintext data.

On the other end of this spectrum are the so called zero-knowledge storage services, in which the encryption process is performed on the client-end, usually with a symmetric encryption algorithm, and only the encrypted data is passed on to the cloud storage provider. Therefore, the cloud service provider technically has the data stored on its premises; it has no way to access the plaintext un-encrypted version of that data because it does not have access to the decryption key. Representative examples of this type of storage providers are services like SpiderOak [3] and Tresorit [4], which support a model where encryption keys are stored at the client side and the cloud service providers do not have the ability to violate the confidentiality of the data they manage. However, the operations that are supported on the data are very limited in this case and they mostly offer a basic create/retrieve interface that supports network backups with no sharing among a community of users.

Compared to current practice the main differences of this use case are as follows:

Firstly, the data owners also own and control the keys used to encrypt their data. They manage these keys by utilising the key management feature offered as a service by the cloud service store.

Secondly, the data owners are able to use policy-based access control mechanisms, which are also available to them as a service from the cloud service store.

Thirdly, an instance of a key management service and the access control service are tightly coupled for each data owner, which allows them to specify key release rules that are applicable only under specific conditions.

Fourthly, the data owners have a choice of different cloud service providers and outsource their data to be stored in different storage mediums depending on factors like application requirements, economics, etc.

Lastly, the data owners have the assurance that their data is stored according to industry standard compliance regimes that cover protection for data-at-rest, key management, and access control across multiple cloud environments.

5.2.4 Objectives

The main objectives of the BT use case are:-

- 1) To provide a protection service for enterprise data over multiple cloud providers;
- 2) To only allow the customers or trusted third parties of their choice to manage the encryption keys in accordance with policy-based access control rules;
- 3) To allow the customers to choose different storage services through a service store that is able to provide an integrated view of multiple cloud storage services in the form of virtual volumes, object stores or Big Data clusters.
- 4) To ensure that the data is being secured in accordance to specific data protection regulations and standards like FIPS, PCI DSS, and HIPAA/HITECH etc. and all activity is logged for audits

5.2.5 Stakeholders

There are four types of stakeholders involved in this use case scenario: the cloud service providers, the data protection service providers, the customer/tenant organisations, and the end-users.

1. The cloud service providers offer data storage services to their consumers, in the forms of block storage, object storage or Big Data clusters etc.
2. The data protection service providers offer the tightly coupled key management and policy-based access control services to the customers and tenants that are subscribed with them, via the cloud service store.
3. The customer or tenant organisations are the consumers of cloud service store that have subscribed to the data protection service in order to securely store and manage their data on multiple cloud storage services.
4. The end-users are the employees or members of a tenant organisation that read and write data on the storage media made accessible to them by their organisations.

5.2.6 Glossary of Acronyms

Acronym	Definition
UC	Use Case
VM	Virtual Machine
KMS	Key Management Server
CSP	Cloud Service Provider
IaaS	Infrastructure as a Service
PaaS	Platform as a Service
SaaS	Software as a Service
DPaaS	Data Protection as a Service
HDFS	Hadoop Distributed File System
REST	Representational State Transfer
SOAP	Simple Object Access Protocol

5.3 Use Cases

The high level business use case is realised by a single core technical use case, which has three different branches acting as sub-use cases. The core technical use case is to offer data protection as a service in a multi-cloud environment to the BT customers. The main components of this system are the data encryption mechanism, access control service and the key management service. The customers will be able to utilise these services via a service store that integrates the protection system and its components with multiple independent cloud service providers.

The division of the main technical use case into three branches is done on the basis of the back-end storage technologies on the multiple cloud providers that are available to the customers and are explained in detail below. The service store will provide the interface for the customers of the DPaaS to access and manage these storage services. As a result of this structure, the sub-use cases will inherit a common encryption, key management and access control system but will have different abstraction layers to cater for the underlying storage mediums.

5.3.1 Use Case: Data Protection as a Service (DPaaS)

This use case will be built on top of underlying services of key management and access control based on key release policies. The service will be offered through a service store where the customer organisation is able to specify what kind of data protection it wants for its end-users.

The use case diagrams are shown below in Figure 12 and Figure 13. The individual use cases are described briefly as follows:

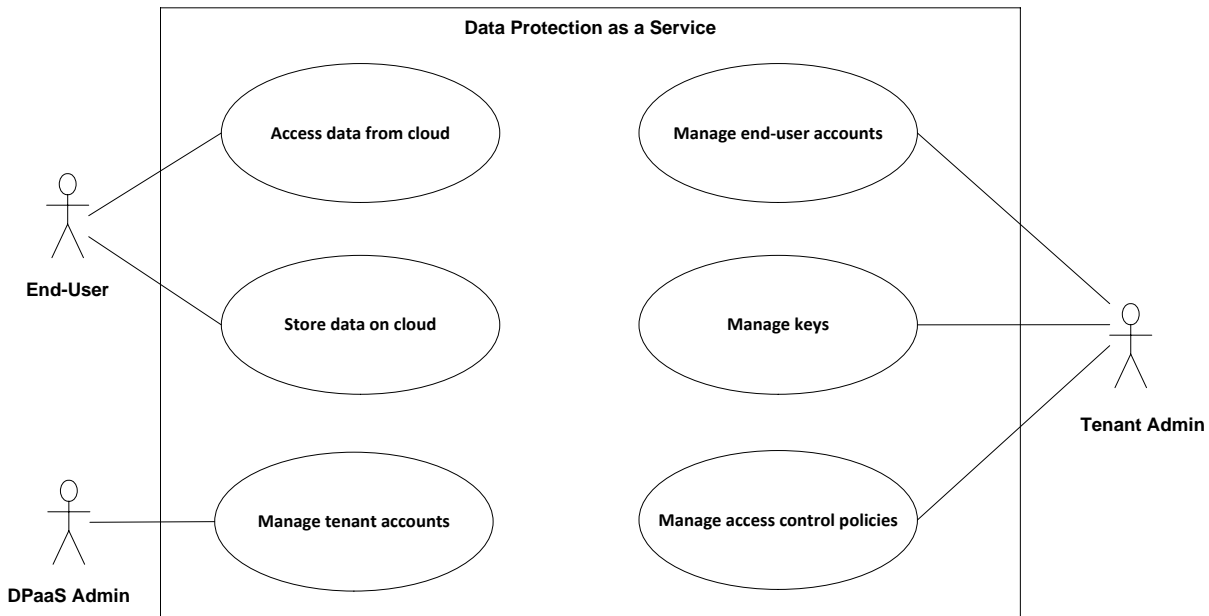


Figure 12. Use case diagram for Data Protection as a Service

- **Access data from cloud storage**
The purpose of this use case is to let the end-user read the encrypted data from the cloud storage service. The decryption process has to be transparent to the end-user.
- **Store data on cloud storage**
The purpose of this use case is to let the end-user write plain data to the cloud storage service. The encryption process has to be transparent to the end-user.
- **Manage end-user accounts**
The purpose of this use case is to let the administrator of the tenant organisation create, remove, and manage accounts for its end-users on the data protection service.
- **Manage keys**
The purpose of this use case is to let the administrator of the tenant organisation create, remove, and assign encryption keys with access control policies on the data protection service. The keys are stored inside a key management server.
- **Manage access control policies**

The purpose of this use case is to let the administrator of the tenant organisation create, remove, and modify access control policies on the data protection service.

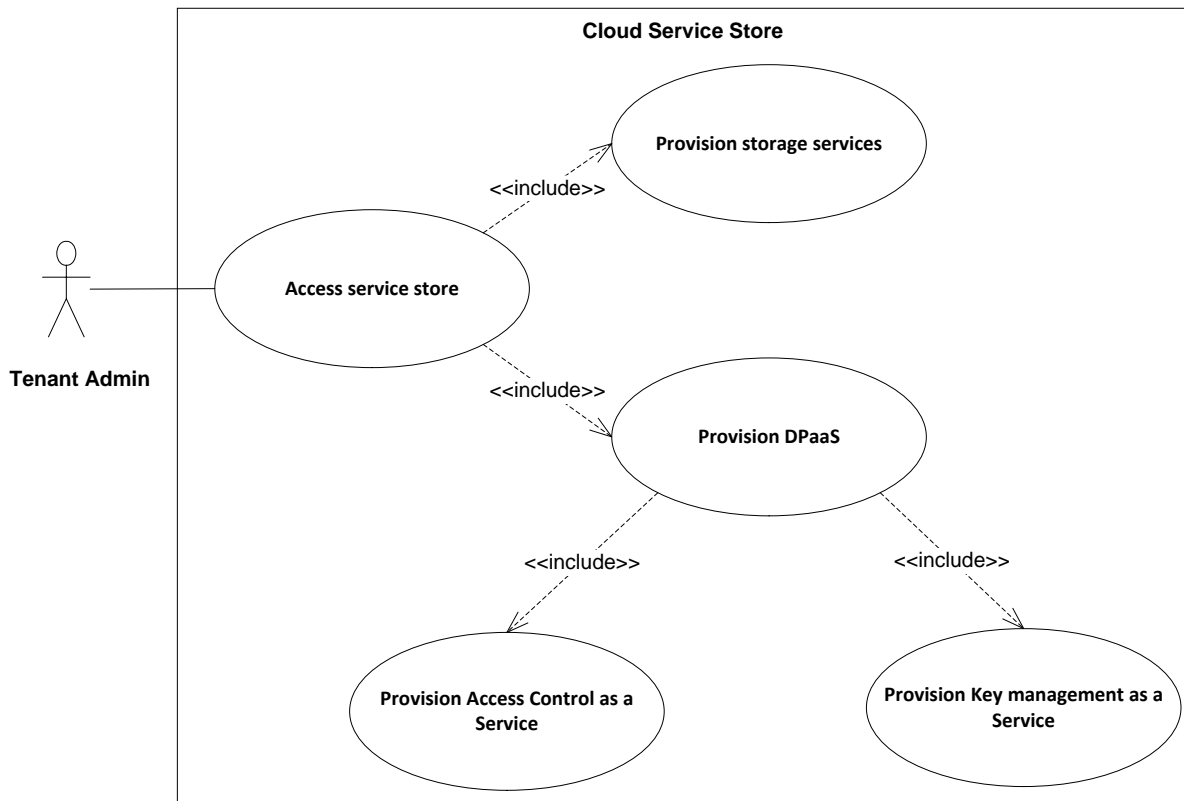


Figure 13. Use case diagram of Cloud Service Store

- Configure and manage service store account

The purpose of this use case is to let the administrator of the tenant organisation access and manage the tenant's cloud service store account. The tenant will be able to specify and control all the attributes related to the cloud storage services and the data protection service through this use case.

- Manage tenant accounts

The purpose of this use case is to let the administrator of the data protection service create, modify or terminate tenant accounts for a customer organisation to start the provisioning or deprovisioning process of the Data Protection as a Service use case.

We branch out the main technical use case in to the following sub-use cases based on the types of storage services being offered to the customer for storing the encrypted content:

5.3.1.1 Store and retrieve data on virtual volumes

The following use case diagram in Figure 14 shows the branch scenario of the core DPaaS use case where the service store offers the customer with the ability to use virtual data volumes as the storage medium of their protected data.

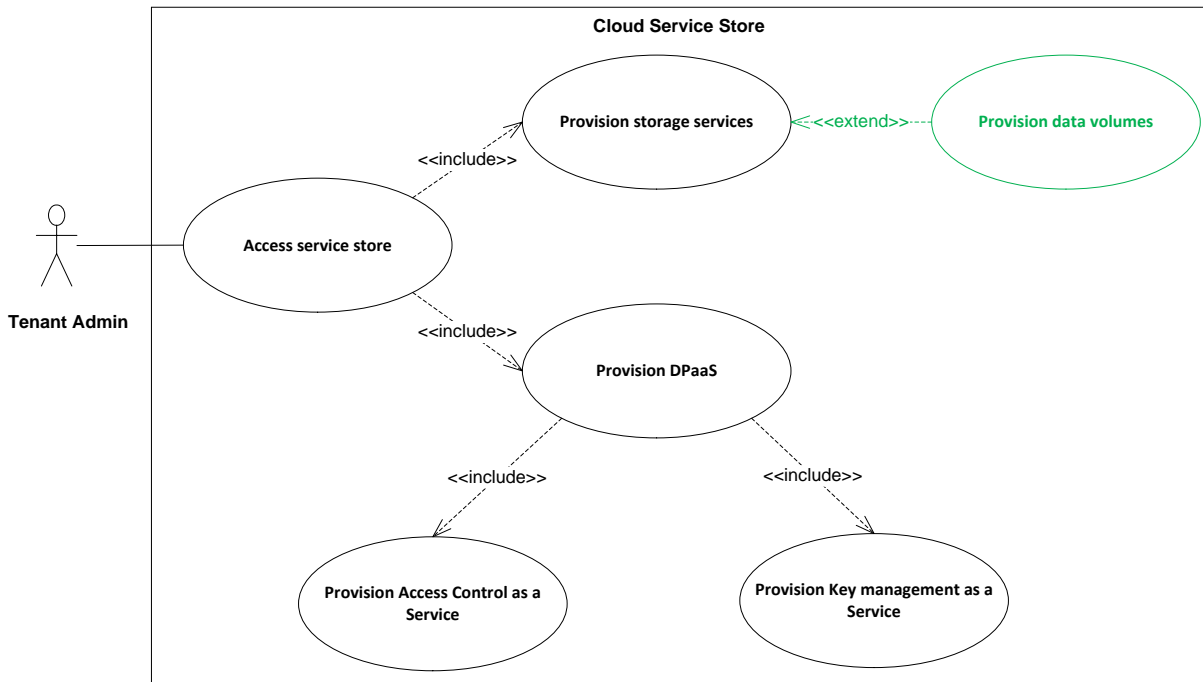


Figure 14. Use case diagram of Cloud Service Store with the Block Storage option

- Actors

There are three actors participating in this use case scenario:-

- The security service's administrator (DPaaS Admin)
- The customer organisation's administrator (Tenant Admin)
- The individual user storing or accessing data from the data volume (End-user)

- Purpose

The purpose of this use case is to provide the end-users with the capability of encrypting the data volumes attached to their VMs on multiple cloud platforms.

- Priority

For this use case, it **MUST** have:

- tenant or 3rd-party managed encryption keys;
- tenant managed policy-based access control of the encryption keys;
- ability to attach and access data volumes with virtual machines on multiple cloud platforms

SHOULD have:

- APIs for integration with a cloud service store;

- FIPS compliant encryption algorithms; and
- Policy based access control defined at the service store

COULD have:

- Low performance overhead for encryption; and
- Easy to use GUI
- Pre-conditions

The pre-conditions for this use case are:

- Encryption keys have been generated on the KMS.
 - The authorised tenants have registered their virtual machines with the system.
 - The authorised tenants have registered their data volumes and their mount points with the system.
 - The authorised tenants have created or associated an access control policy for their mount points.
 - Only authorised end-users or their agents are able to request the encryption keys.
- Post-conditions

The post-conditions for this use case are:-

- The data volume attached to the authorised VM has been encrypted.
- End users can access and store data transparently on the protected data volume.

5.3.1.2 Store and retrieve data on object storage

The following use case diagram in Figure 15 shows the branch scenario of the core DPaaS use case where the service store offers the customer with the ability to use object storage services on multiple cloud platforms as the storage medium of their protected data.

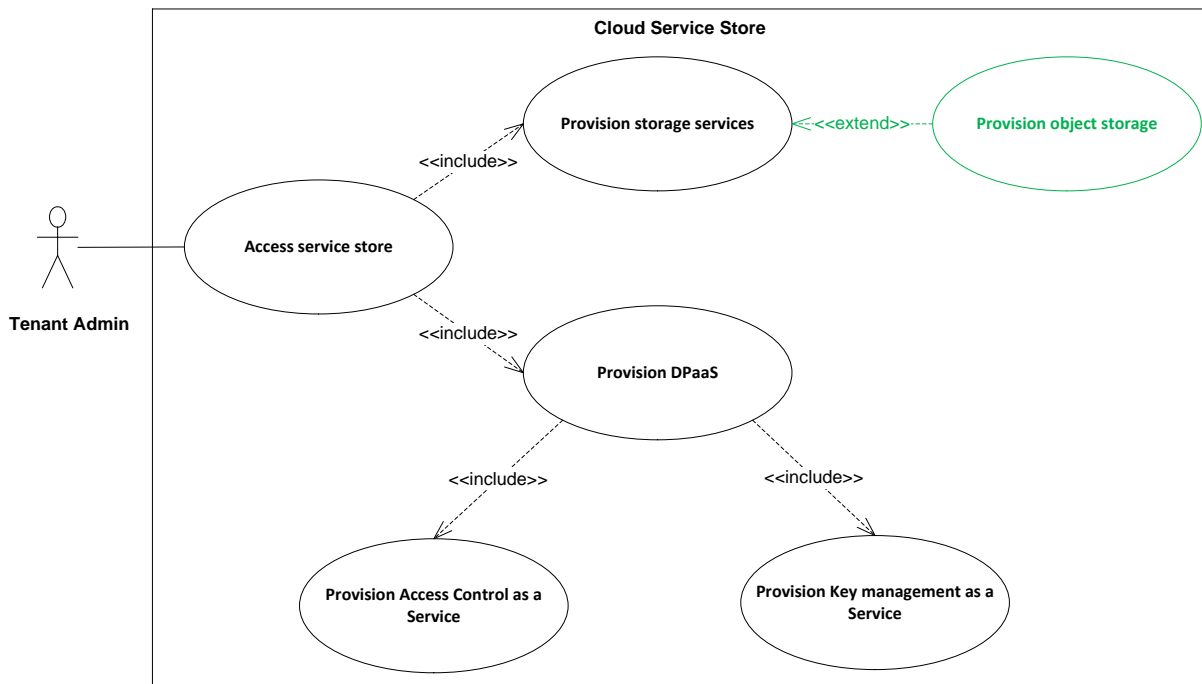


Figure 15. Use case diagram of Cloud Service Store with the Object Storage option

- Actors

There are three actors participating in this use case scenario:-

- The security service's administrator (DPaaS Admin)
- The customer organisation's administrator (Tenant Admin)
- The individual user storing or accessing data from the object storage (End-user)

- Purpose

The purpose of this use case is to provide the end-users with the capability of encrypting and decrypting their data using the object storage as the storage medium on multiple cloud platforms.

- Priority

For this use case, it **MUST** have:

- tenant or 3rd-party managed encryption keys;
- tenant managed policy-based access control of the encryption keys;
- access to the object storage services of multiple cloud platforms

SHOULD have:

- APIs for integration with a cloud service store;
- FIPS compliant encryption algorithms; and
- Policy based access control defined at the service store

COULD have:

- Low performance overhead for encryption; and
- Easy to use GUI

- Pre-conditions

The pre-conditions for this use case are:

- Encryption keys have been generated on the KMS.
- The authorised tenants have registered their object store entry points with the system.
- The authorised tenants have registered their object buckets with the system.
- The authorised tenants have created or associated an access control policy for their data objects.
- Only authorised end-users or their agents are able to request the encryption keys.

- Post-conditions

The post-conditions for this use case are:-

- The data object to be stored on the object store has been encrypted.
- End users can access and store objects transparently on their object store.

5.3.1.3 Store and retrieve data on Big Data service

The following use case diagram in Figure 16 shows the branch scenario of the core DPaaS use case where the service store offers the customer with the ability to use Big Data clusters as the storage medium of their protected data.

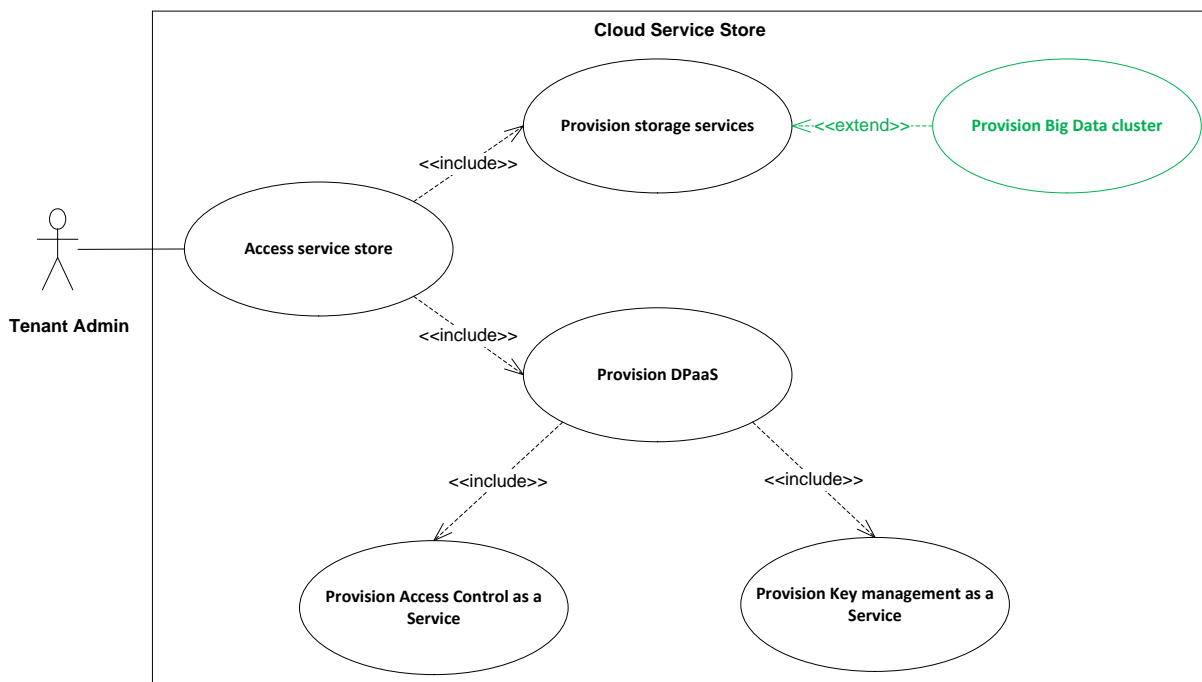


Figure 16. Use case diagram of Cloud Service Store with the Big Data option

- Actors

There are three actors participating in this use case scenario:-

- The security service's administrator (DPaaS Admin)
- The customer organisation's administrator (Tenant Admin)
- The individual user storing or accessing data from the Big Data services (End-user)

- Purpose

The purpose of this use case is to provide the end-users with the capability of encrypting and decrypting their data using a Big Data service (e.g., a Hadoop cluster) as the storage backend on multiple cloud platforms.

- Priority

For this use case, it **MUST** have:

- tenant or 3rd-party managed encryption keys;
- tenant managed policy-based access control of the encryption keys;
- access to the Big Data services of multiple cloud platforms

SHOULD have:

- APIs for integration with a cloud service store;
- FIPS compliant encryption algorithms; and
- Policy based access control defined at the service store

COULD have:

- Low performance overhead for encryption; and
- Easy to use GUI

- Pre-conditions

The pre-conditions for this use case are:

- Encryption keys have been generated on the KMS.
- The authorised tenants have registered their HDFS mount points with the system.
- The authorised tenants have created or associated an access control policy for their HDFS mount points.
- Only authorised end-users or their agents are able to request the encryption keys.

- Post-conditions

The post-conditions for this use case are:-

- The data to be stored on the HDFS has been encrypted.
- End users can access and store objects transparently on their HDFS store.

5.3.1.4 Sequence of events

The following figure shows the expected set of interactions between the different actors and the system in the case of a customer that wants to provision a data volume from a cloud service and encrypt it using the encryption key obtains from its key management server. The sequence diagram below in Figure 17 provides the high level description:-

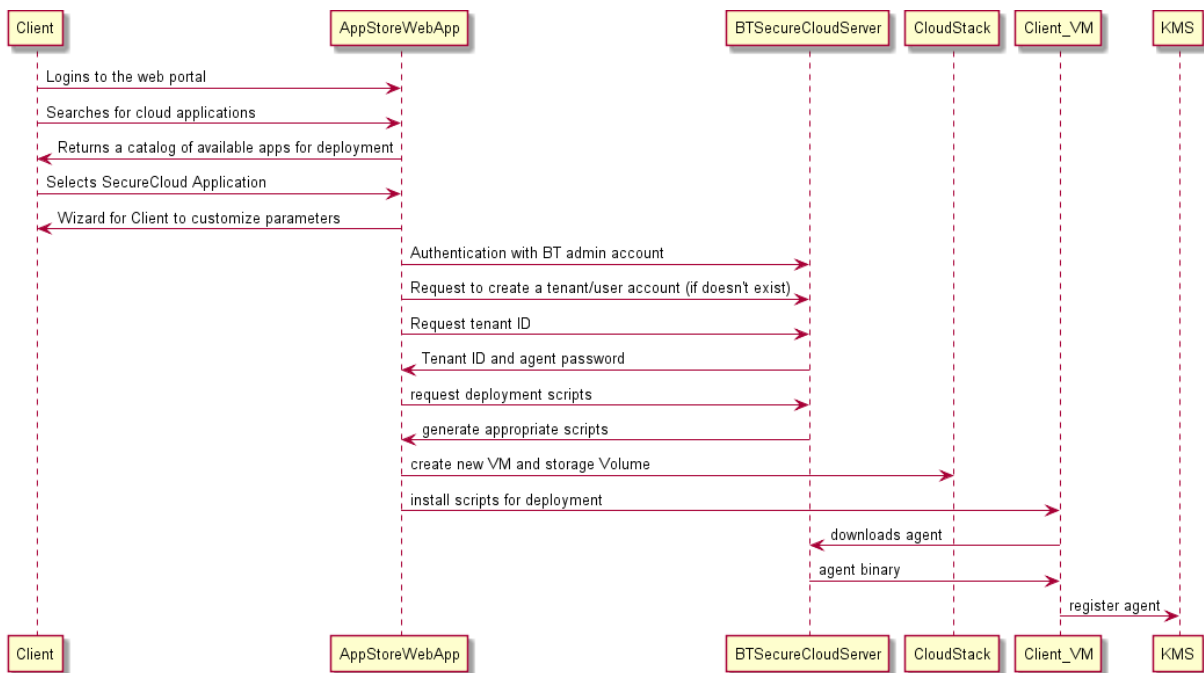


Figure 17. Sequence diagram for the DPaaS use case

The client using a cloud service store (AppStoreWebApp) selects the Data Protection as a Service (BTSecureStorage) product and through a guided wizard instructs the service store to deploy a VM with SecureStorage agent installed and configured. The client using the graphical interface provided by the service store will set the requirements (these are high level requirements: can be selection of the type of VM and size of the storage volume that is going to be used as an encrypted volume). The service store having all these parameters will contact the BTSecureCloudServer to get the deployment and installation script; these scripts will be installed to the client VM and will ensure the installation and configuration of SecureStorage agent.

5.3.1.5 Storyboard

We depict the storyboard of the DPaaS use case in the form of mock screen shots of a data protection service, as shown below in a series of figures. The figures show a conceptual customer journey for a tenant trying to provision secure data volumes to its end-users. Each figure corresponds to a step of this journey, which can be summarised as:

1. The tenant logs in to the cloud service store and chooses the cloud service providers offering the block storage service (Figure 18).
2. The tenant chooses to provision the “Data Protection” service on top of its storage service (Figure 19).
3. The tenant provisions the virtual machines with which the block storage devices will be attached and encrypted (Figure 20).

4. The tenant provides the mount point through which the end-user will be able to access and use the secure data volume (Figure 21).
5. The tenant will create or provision keys to the KMS which will be used to encrypt the data volume (Figure 22).
6. The tenant will create or provision the access control policy which will be used to govern the release of encryption keys (Figure 23).

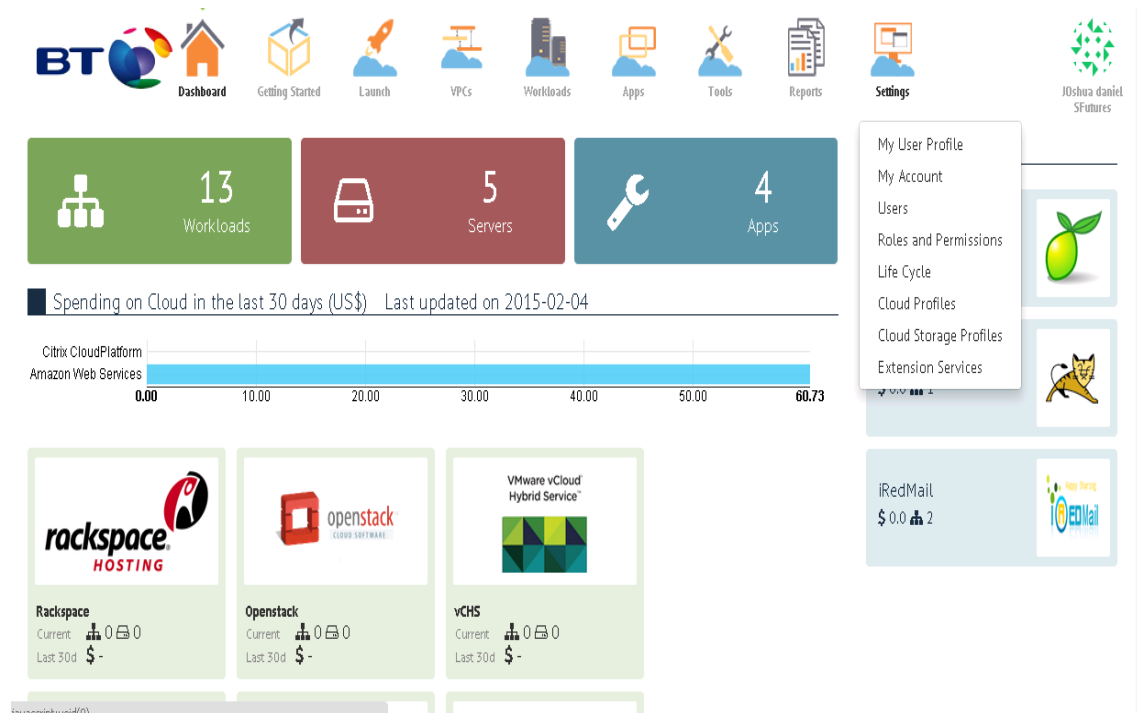
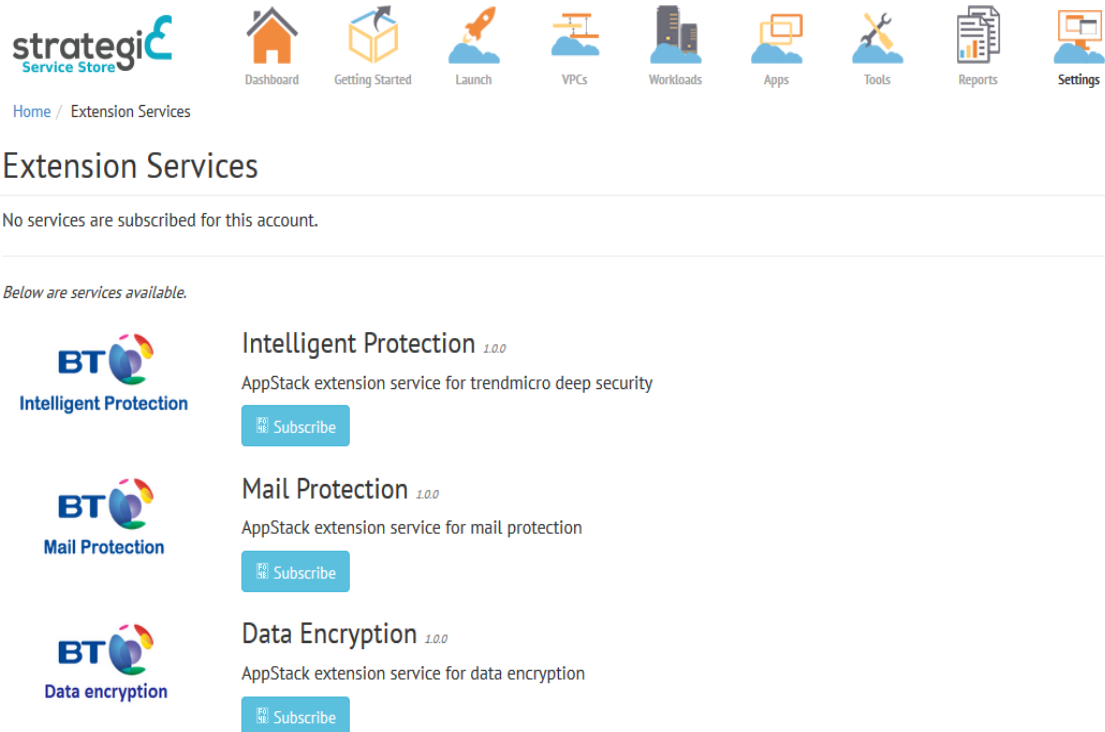


Figure 18. Step 1 – Choosing a cloud service provider through a service store




strategi€ Service Store

Home / Extension Services

Extension Services

No services are subscribed for this account.


Below are services available.



Intelligent Protection 1.0.0

AppStack extension service for trendmicro deep security


Subscribe



Mail Protection 1.0.0

AppStack extension service for mail protection

Subscribe

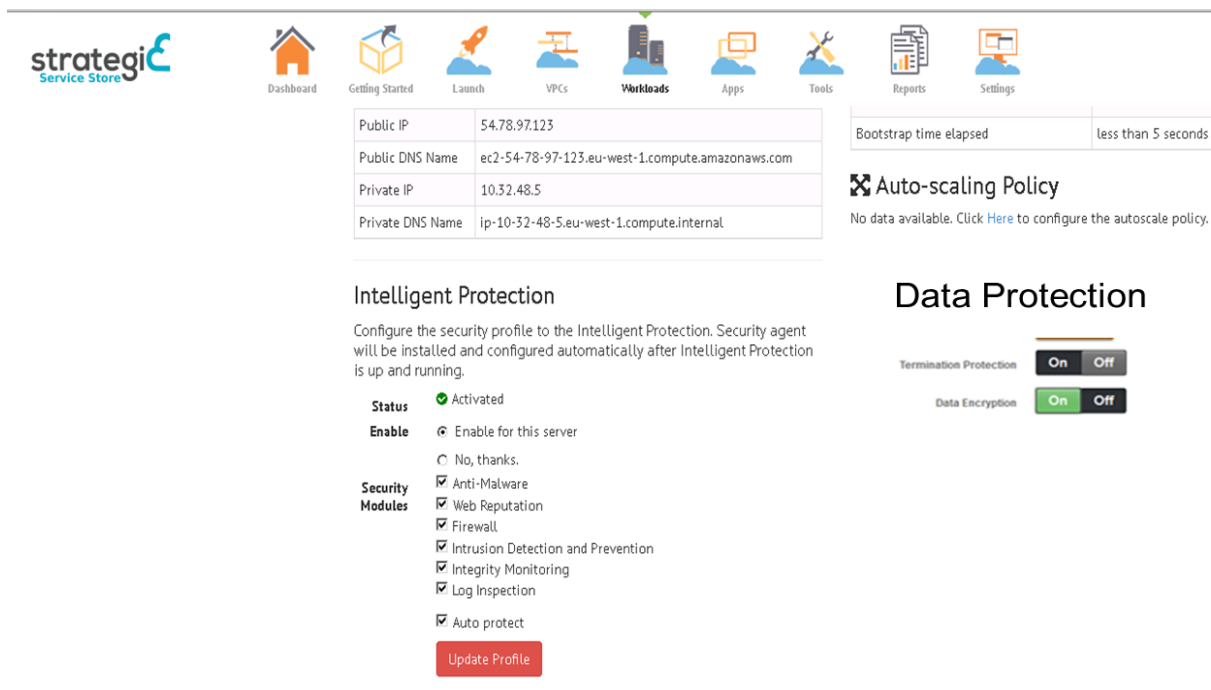


Data Encryption 1.0.0

AppStack extension service for data encryption

Subscribe

Figure 19. Step 2 - Select the 'Data Encryption' service option



strategi€ Service Store

Workloads

Public IP	54.78.97.123
Public DNS Name	ec2-54-78-97-123.eu-west-1.compute.amazonaws.com
Private IP	10.32.48.5
Private DNS Name	ip-10-32-48-5.eu-west-1.compute.internal

Bootstrap time elapsed: less than 5 seconds

Auto-scaling Policy

No data available. Click [Here](#) to configure the autoscale policy.

Intelligent Protection

Configure the security profile to the Intelligent Protection. Security agent will be installed and configured automatically after Intelligent Protection is up and running.

Status ☒ Activated

Enable ☒ Enable for this server

☐ No, thanks.

Security Modules

- ☒ Anti-Malware
- ☒ Web Reputation
- ☒ Firewall
- ☒ Intrusion Detection and Prevention
- ☒ Integrity Monitoring
- ☒ Log Inspection
- ☒ Auto protect

Update Profile

Data Protection

Termination Protection: ☒ On ☐ Off

Data Encryption: ☒ On ☐ Off

Figure 20. Step 3 - Provision virtual machine with the Data Protection service

strategize Service Store

Home / My Workloads / Microsoft Office

Microsoft Office eu-west-1

Overview Applications Security Backup Volumes Graph Monitoring

+ Add Volume Refresh

Size	Created at	Type	Status	Action
80 GB	2015-02-19 15:33:22 UTC	root	active	

Add New Protection

Size (GB)

D:\My Documents\

Encrypt or Cancel

All data in this folder will be encrypted.

Figure 21. Step 4 - Specify a mount point for a data volume that requires encryption

Agent Keys

Search

Name Contains

Select All View 20

Add Delete

Selected	UUID	Name	Algorithm	Key Type	Creation Date
<input type="checkbox"/>	02-92	SCS4FireKey	AES128	Cached on Host	Oct 16, 2014
<input type="checkbox"/>	02-143	ali	ARIA128	Cached on Host	Apr 27, 2015
<input type="checkbox"/>	1	clear_key	CLEAR	Stored on Server	
<input type="checkbox"/>	02-107	joshua	AES256	Cached on Host	Feb 02, 2015
<input type="checkbox"/>	02-144	pramod	ARIA256	Cached on Host	Apr 27, 2015
<input type="checkbox"/>	02-145	rob	AES128	Cached on Host	Apr 27, 2015
<input type="checkbox"/>	02-146	theo	3DES	Stored on Server	Apr 27, 2015

Add Delete

Figure 22. Step 5 - Select an encryption key to be associated with the data

Figure 23. Step 6 - Configure a security policy to govern the key release

Figure 18 to Figure 23 will be the same in case of all branches of the core use case, except Figure 21, which will be different in each use case, depending on the storage service chosen by the customer. In this particular instance, we assume that the customer has chosen to encrypt their data on a data volume attached to a particular VM.

5.4 Requirements for Data Protection as a Service use case

5.4.1 Introduction

5.4.1.1 Purpose

The scope of this solution is to offer data protection as a service to customers who want to outsource the storage of their data in a multi-cloud environment. Therefore, the core parts of this solution are the data encryption mechanisms, key management and access control based on security policies, as well as the seamless integration of these components to provide the customers with a standalone security service. The customers will apply data protection via a service store that integrates the protection service with storage back-ends/medium on multiple independent cloud service providers.

5.4.1.2 Conventions

None at the moment.

5.4.1.3 Major relevant ESCUDO-CLOUD dimensions

ESCUDO-CLOUD considers four different dimensions [6] that help provide data owners with the combination of security and flexibility when trying to outsource their data resources to the cloud. These dimensions are shown in Table 1.

The Data Protection as a Service use case operates in all of these four dimensions. The security challenges addressed by different components of this solution in the stated ESCUDO-CLOUD dimensions are given below:-

- Security properties

The core responsibility of the data protection service is to ensure the confidentiality, integrity and availability of the customer's data-at-rest, which is enforced by using client-side encryption

approach. Furthermore, the access and usage of the key management and access control features should also conform to these security properties.

- **Sharing requirements**

We have to cater for the two situations in this use case regarding the sharing of customer data. The first is the simpler situation where only the customers have access to the encryption keys and thus no one else, especially the cloud service providers, have the ability to read the plaintext data. The second is a more complex situation where the customers can make their data available to others by sharing the keys and modifying access control policies.

- **Access requirements**

In this dimension we deal with the issues of providing the customers with the means to access and retrieve their data from different types of cloud-based storage services, e.g., block storage, object storage and Big Data services. This also includes the aspects of controlling access to the data through policy based security rules that enforce the release of keys.

- **Cloud architectures**

This use case operates in a multi-cloud environment, empowering the customers with the ability of selectively using different cloud service providers depending on factors of their choice e.g., trust, economy etc.

5.4.2 Initial solution architecture in the context of the use-cases

The solution provides data protection as a service in a multi-cloud environment to enterprise customers. The main components of this system are the data encryption mechanism, access control service and the key management service. The customers will be able to utilise these services via a service store that integrates the protection system and its components with multiple independent cloud service providers.

The main benefit to the customers is the ability to outsource their data storage while still maintaining access control over it. This is achieved thanks to the secure key management as only the customers possess the encryption keys. They also have the ability to regulate access to their data through policy based enforcement of rich access control attributes. They will also have the assurance that their data will remain protected from the untrusted cloud service providers. The customers will be provided with the choice of selecting from multiple CSPs. In addition, by retaining control of key management the customers have more flexibility in meeting regulatory and privacy requirements and ensuring data confidentiality and secure access.

5.4.3 Overall Description

5.4.3.1 Solution Perspective

The architecture for the Data Protection as a Service use case is given in Figure 24.

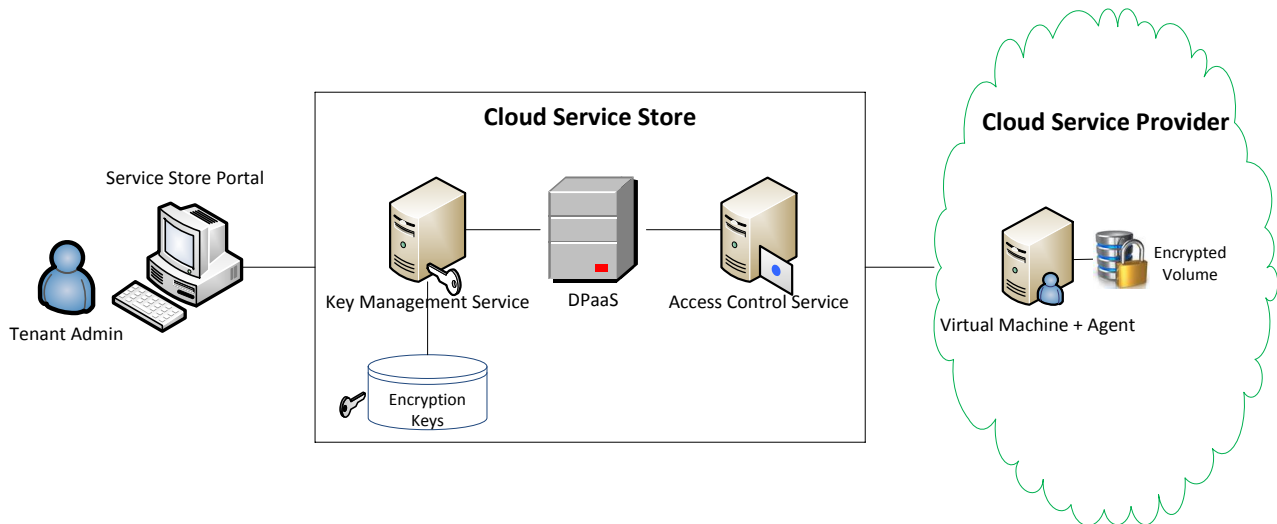


Figure 24. Initial solution architecture of the Data Protection as a Service use case to encrypt data volumes attached to virtual machines

The cloud service store is used to provision the overall service to the tenants. Each tenant will get a compartmentalised access to the service store and the data protection service, as discussed above. The tenant will be able to define the access control and key release policies via the service store or the KMS interface and also associate keys with those policies. The service store also has the ability to install and configure the encryption agent software on virtual machines on different supported cloud platforms.

After successful provisioning by the service store, the agent on the machines can use pre-boot authentication to identify itself to the Key Management Server. If the agent successfully passes the pre-boot authentication, then the Key Management Server issues the key necessary to decrypt the boot volume. After configuration, the agent submits an encryption key request to the Key Management Server.

This request may be manually or automatically approved, depending on which policy is assigned to the machine on which the agent is installed. Once the Key Management Server issues the encryption key, the agent can use the key to provision (encrypt) any devices attached to the machine.

5.4.3.2 Solution Functions

The list of the major functions the solution must perform is as follows:

- Multi-tenant key management
- Multi-tenant policy-based access control
- Integrated/Tightly coupled provisioning of key management and access control services through a service store
- Provisioning of encryption agents or plug-ins that are able to utilise specific algorithms to perform primitive encryption functions

- Abstract provisioning of different storage services on multiple cloud service providers through the service store
- Encryption of data on block storage devices
- Encryption of data on object storage services
- Encryption of data on Big Data services

5.4.3.3 Actors and Characteristics

Following is the list of the actors that will use this solution, as well as their salient characteristics:-

- Service store administrator

Characteristics: This actor will enable access to the resources of multiple cloud service providers for the tenant organisations, via a web based user interface. It will also offer the tenants with the starting point of the provisioning process of the Data Protection as a Service. Lastly, it will also provide the tenants with the interface to provision and manage the storage service (block storage, object storage, etc.) hosted on multiple cloud platforms.

- DPaaS administrator

Characteristics: This actor will use the solution to create tenant accounts that will be used to access the resource of the DPaaS.

- Tenant administrator

Characteristics: This actor will use the solution to create and manage the encryption keys. It will also create and manage access control policies that will govern the release of the encryption keys to trusted encryption agents and plug-ins. It will also point to the reference locations where the encrypted data will be stored.

- End-User

Characteristics: This actor will use the solution in transparently to store and retrieve data from secured data locations.

5.4.3.4 Operating Environment

Most of the components of the Data Protections as a Service solution will be operating primarily as services running on virtual machines in the BT Compute Cloud platform, which is based on the Cloud Platform which is the commercial variant of the CloudStack infrastructure as a service (IaaS) solution. The underlying hypervisor platforms of the IaaS solution can be either Xen or VMWare. The virtual machines will run the operating systems based on a recent open source Linux based distribution e.g., Ubuntu 14.04 or CentOS 7 etc. The service should enable protection of data on IaaS including Cloud Platform, OpenStack, Amazon AWS and S3 compatible object store solutions (e.g. Castor by Caringo).

5.4.3.5 Design and Implementation Constraints

In order to provide data protection as a service in a multi-cloud environment to enterprise customers we will have to work within the following set of design and implementation constraints:



- There has to be a tight integration between a customer's instance of the key management service and access control service.
- The key management and access control services need to be hosted in a secure environment (either on customers premises or a trusted service store) and only the customer should have access to the key store and its associated policy store based on robust authentication mechanisms.
- There has to be automatic provisioning and configuration of the encryption agents or plug-ins on the data protection service's back-ends via the service store.
- Encryption keys are securely deleted after decommissioning of the protection service and are never left with the cloud service provider at rest.
- No data should be lost when performing encryption or decryption operations on data volumes, object stores or HDFS clusters.
- Customers should be able to unsubscribe from the data protection service without suffering loss of the protected data.
- A secure communication protocol needs to be used between the key management service and the encryption agents and plug-ins.
- The encryption agents or plug-ins need to support the operating systems offered by the service store.

5.4.3.6 Assumptions and Dependencies

The assumptions and dependencies regarding the components of the Data Protection as a Service solution are listed below:-

- The service store offers the capability of provisioning the encryption agents and plug-ins offered by the DPaaS.
- It is able to deploy, configure and manage these agents and plug-ins on the tenant owned virtual machines and gateways hosted on different cloud service provider.
- The core encryption operations will rely on commercial and accredited technologies.
- The data encryption and decryption process will be transparent to the end-users of the solution.
- The service store offers an independent access control as a service that enforces the data-at-rest encryption of the data stored in the infrastructure of multiple cloud platforms.
- Key management as a service feature is a part of the service store offerings.
- The core key management operations will rely on commercial and accredited technologies.
- Third party key management services cannot be used due to the tight integration between the key management service and the access control service.
- The tenant trusts the service store and the data protection service being offered through it.
- The tenant places limited trust on the cloud service providers that will eventually host the stored data.

- The data protection solution caters for the Honest-but-Curious model, in which the cloud service providers store the outsourced data without tampering it, and retrieve the data on request from the owner without any malicious modifications.
- The cloud service providers may try to look in to or analyse the stored data.

5.4.4 External Interface Requirements

5.4.4.1 User Interfaces

All of the user interfaces will be web-based and accessible to users via a web browser upon submission of correct credentials. The following components of the solution require user interfaces:-

- Data Protection as a Service
 - A login interface for the DPaaS administrator
 - A dashboard interface for the DPaaS administrator to view, add, delete and modify tenant accounts
 - A login interface for the tenant administrators to login to their instance of the service
 - A dashboard interface for the tenant administrators to view the status of their protected data
 - An interface for the tenant administrators to manage the keys
 - An interface for the tenant administrators to manage the access control policies
 - An interface for the tenant administrators to associate/bind data-to-be-encrypted with access control policies.
- Key Management as a Service
 - A login interface for the tenant administrator
 - A dashboard interface for the tenant administrator to view, add, delete and modify encryption keys
- Access Control as a Service
 - A login interface for the tenant administrator
 - A dashboard interface for the tenant administrator to view, add, delete and modify security rules and policies
 - An interface to couple encryption policies with specific keys and their operations
- Service Store
 - A login interface for the DPaaS administrator
 - A login interface for the tenant administrator
 - An interface to add, modify and delete cloud service providers to the service store

5.4.4.2 Hardware Interfaces

The solution will be hardware independent.

5.4.4.3 Software Interfaces

We can divide the software interfaces required by this solution into three categories:-

- HTTP based interfaces for the data owners (tenants)
- SOAP/REST based APIs for the service store
- SOAP/REST based APIs for the cloud service providers

5.4.4.4 Communications Interfaces

The communication between the data owners (tenants) and the service store will take place through a web interface (HTML), using any current web browsers as the communication client.

This communication will be secured with the help of the SSL/TLS standard protocol suite and the tenant will be able to use the data protection service only after proper authentication and authorisation processes have taken place.

5.4.5 System Features

5.4.5.1 Key Management (KM)

5.4.5.1.1 Description and Priority

This feature enables the tenants to manage the keys used to encrypt their data. This feature is offered as a service by the cloud service store.

5.4.5.1.2 Stimulus/Response Sequences

An authorized user (tenant admin) can perform the following actions using this feature:-

- Send a key generation request with the following attributes:-
 - Key name
 - Generation algorithm
 - Key string (optional)
 - Expiry date (optional)
- Modify keys
- Delete keys

5.4.5.1.3 Functional Requirements

REQ-UC3-KM-1: Each tenant should be provisioned with an instance of a key management service from the cloud service store. Each tenant is given a unique identifier at the time of the tenant account creation and the same identifier is used across the components of the data protection solution to identify and reference a tenant. Each tenant account has further (at least one)

administrator accounts that are used to manage the tenants instances of key management, access control, and data storage and encryption services.

REQ-UC3-KM-2: The tenants should be able to generate, modify and remove keys from their key management service instance. The modification is limited to specific attributes of a key e.g., the expiry date.

REQ-UC3-KM-3: The key management service should be able to offer different key types and generation algorithms to each tenant, e.g., AES128, AES256, 3DES etc.

REQ-UC3-KM-4: Only the tenants should be able to create and manage the keys. This should be enforced by safeguarding access to the tenant's instance of the key management service with a robust authentication and authorisation mechanism.

REQ-UC3-KM-5: The cloud service providers should have no access or visibility of the tenants' keys. This should be enforced by ensuring that the key store used by the tenant's key management service is not hosted on untrusted platforms or cloud service providers. Furthermore, keys can only be release from the key store if they pass the access control checks associated with them.

REQ-UC3-KM-6: The tenants should be able to cache their keys on trusted virtual machines or gateways in order to outsource or improve performance of the encryption and decryption process. The encryption and decryption process should be carried out by a software agent or plug-in that has been provisioned on the trusted virtual machine by the service store. The keys should only be cached for the duration of the encryption or decryption process and should not be present on the virtual machine or the gateway while the data is at rest.

5.4.5.1.4 Relevant ESCUDO-CLOUD dimensions

The relevant ESCUDO-CLOUD dimensions are described below:-

- Security properties

Both the access and usage of the key management service and the encryption keys by the tenant should conform to the confidentiality, integrity and availability properties.

- Sharing requirements

Only the tenant should have access to the encryption keys and thus no one else, especially the cloud service providers, have the ability to access and use the keys. An exception exists in the scenario where the tenant can share the keys with encryption agents on virtual machines and gateways.

- Access requirements

The tenant is able to upload keys to the key management service, but the only way to release a key is through the approval of policy-based security rules associated with that key.

- Cloud architectures



This feature operates in a multi-cloud environment, where the key management service can be deployed on tenant's own platform or any cloud platform trusted by the tenant.

5.4.5.2 Access Control (AC)

5.4.5.2.1 Description and Priority

This feature enables the tenants to manage the access control rules and policies used to govern the release of encryption keys. This feature is offered as a service by the cloud service store.

5.4.5.2.2 Stimulus/Response Sequences

An authorized user (tenant admin) can perform the following actions using this feature:-

- Create access control policies with the following attributes:-
 - Security rules (specification of conditions to be checked)
 - Key selection rules (keys to be released and operations permitted with those keys if the conditions are met)
- Modify access control policies
- Delete access control policies

5.4.5.2.3 Functional Requirements

REQ-UC3-AC-1: Each tenant should be provisioned with an instance of an access control service from the cloud service store. Each tenant is given a unique identifier at the time of the tenant account creation and the same identifier is used across the components of the data protection solution to identify and reference a tenant. Each tenant account has further (at least one) administrator accounts that are used to manage the tenants instances of key management, access control, and data storage and encryption services.

REQ-UC3-AC-2: The tenants should be able to create, delete and modify access control policies from their instance of the access control service.

REQ-UC3-AC-3: The access control service should be able to offer use of different system and data attributes for the construction of a security rule, e.g., filesystem, user, application, and time attributes. Each access control policy can be comprised of a collection of security rules to offer and modular approach.

REQ-UC3-AC-4: Only the tenants should be able to create and manage their access control policies. This should be enforced by safeguarding access to the tenant's instance of the access control service with a robust authentication and authorisation mechanism.

REQ-UC3-AC-5: The cloud service providers should have no access or visibility of the tenants' keys. This should be enforced by ensuring that the policy manager and the policy database used by the tenant's access control service is not hosted on untrusted platforms or cloud service providers.

REQ-UC3-AC-6: All data protection operations should be governed by access control policies by either approving or denying access to the required keys. The client-side encryption agent or plug-in should be able to request the access control service for issuing it with the encryption key. The request has to include the attributes that are present in the security policy so that the access control service can evaluate the request conditions and take action to either release the key or deny the request.

REQ-UC3-AC-7: The access control service of tenants should be tightly coupled with their key management service, such that no key can be utilised without an approving access control policy.

5.4.5.2.4 Relevant ESCUDO-CLOUD dimensions

The relevant ESCUDO-CLOUD dimensions are described below:-

- Security properties

Both the access and usage of the access control service by the tenant should conform to the confidentiality, integrity and availability properties.

- Sharing requirements

Only the tenant should have access to its instance of the access control service and thus no one else, especially the cloud service providers, have the ability to access and view the policies.

- Access requirements

The tenant is able to upload and download policies to and from the access control service, as well as write/construct policies directly on the service.

- Cloud architectures

This feature operates in a multi-cloud environment, where the access control service can be deployed on tenant's own platform or any cloud platform trusted by the tenant.

5.4.5.3 Service Orchestrator (SO)

5.4.5.3.1 Description and Priority

This feature provisions the main components of the data protection solution to the tenants. It also provisions the access to different storage services to the tenants for the storage of their encrypted data. It also abstracts the access to multiple cloud service providers for the tenants.

5.4.5.3.2 Stimulus/Response Sequences

The service store administrator can perform the following actions using this feature:-

- Create, modify or delete profiles of multiple cloud service providers on the service store
- Create, modify or delete access/reference points of different storage services for each tenant

An authorized user (tenant admin) can perform the following actions using this feature:-

- Provision the data protection solution
- De-provision the data protection solution

5.4.5.3.3 Functional Requirements

REQ-UC3-SO-1: Each tenant should be provisioned with a cloud service store account. Each tenant is given a unique identifier at the time of the tenant account creation and the same identifier is used across the components of the data protection solution to identify and reference a tenant. Each tenant account has further (at least one) administrator accounts that are used to manage the tenants instances of key management, access control, and data storage and encryption services.

REQ-UC3-SO-2: The service store should provide the tenants with access to the storage services of multiple cloud service providers. This should be done by creating transparent access profiles of each cloud service provider so that the tenant does not have to perform cloud provider specific authentication and access procedures.

REQ-UC3-SO-3: The service store should be able to offer block storage service to the tenants. The tenants should be able to specify the size, format etc. of the block storage and attach it as a data volume with their virtual machines.

REQ-UC3-SO-4: The service store should be able to offer object storage service to the tenants. The tenants should be able to specify the object buckets' names and other properties and be able to provision it for the use of the end-users.

REQ-UC3-SO-5: The service store should be able to offer Big Data storage service (HDFS) to the tenants. The tenants should be able to provision the configured HDFS cluster for the use of the end-users.

REQ-UC3-SO-6: The tenants should be able to enable or disable the use of data protection service on the storage service of their choice.

REQ-UC3-SO-7: The service store should be able to offer key management as a service to the tenants. This is a pre-requisite for the KM feature described above.

REQ-UC3-SO-8: The service store should be able to offer access control as a service to the tenants. This is a pre-requisite for the AC feature described above.

5.4.5.3.4 Relevant ESCUDO-CLOUD dimensions

The relevant ESCUDO-CLOUD dimensions are described below:-



- Cloud architectures

This feature operates in a multi-cloud environment, where the service store can be deployed on any cloud platform and it further on provides a single abstraction layer for the consumption of the storage services of multiple cloud service providers.

5.4.5.4 Data Encryption (DE)

5.4.5.4.1 Description and Priority

This feature provisions the core encryption components of the data protection solution to the tenants and the end-users.

5.4.5.4.2 Stimulus/Response Sequences

The service store administrator can perform the following actions using this feature:-

- Create, modify or delete profiles of multiple cloud service providers on the service store
- Create, modify or delete access/reference points of different storage services for each tenant

An authorized user (tenant admin) can perform the following actions using this feature:-

- Provision the data protection solution
- De-provision the data protection solution

5.4.5.4.3 Functional Requirements

REQ-UC3-DE-1: The core encryption process should only be controlled and managed by the tenant. It should be transparent to the end-user and only the tenant should be able to specify the target storage services where the encrypted data has to be stored.

REQ-UC3-DE-2: The tenant should be able to deploy and manage the core encryption process on trusted virtual machines or gateways as an agent or plug-in. This should be done via the use of the service store's interface to the associated cloud service provider where the target virtual machine or gateway is hosted.

REQ-UC3-DE-3: The core encryption process should be FIPS 140 compliant.

REQ-UC3-DE-4: The encryption agent or plug-in should be able to access the tenant's key management service and access control service. This should be done by enabling it to connect to these services over an encrypted channel using protocols like SSL/TLS.

REQ-UC3-DE-5: The keys should only be released to the encryption agent or plug-in upon approval of an access control policy. The encryption operation should only be performed after the

policy authorising the encryption has been met and the associated key has been released by the access control service

5.4.5.4.4 Relevant ESCUDO-CLOUD dimensions

The relevant ESCUDO-CLOUD dimensions are described below:-

- Security properties

The encryption algorithms and mechanisms used by the tenant should conform to the FIPS 104 standard and hence provide the necessary confidentiality, integrity and availability properties.

- Sharing requirements

The encrypted data can be shared with anyone by the tenant but it will not be readable in absence of the encryption keys that only the tenant controls and possess. In order to share the data with third parties, the tenant will also have to share the associated encryption key.

- Access requirements

The tenant is able to upload and download encrypted data to and from the virtual volume, object store and the HDFS cluster. In the case of virtual volume storage, the data can be decrypted on the virtual machine but only after the release of the associated encryption key through the approval of policy-based security rules bound with that key.

- Cloud architectures

This feature operates in a multi-cloud environment.

5.4.6 Requirements Catalogue

The summarized security requirements for the Data Protection as a Service use case are given in the following table:-

Table 4. Requirements Catalogue for Use Case 3

Requirement Reference #	Requirement Description	ESCUDO-CLOUD dimension	Priority	Dependencies on other Requirement	Relevant Core Work Package
REQ-UC3-KM-1	Each tenant should be provisioned with an instance of a key management service from the cloud service store	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Access by data owners - Multi clouds 	High	REQ-UC3-AC-7 REQ-UC3-SO-1 REQ-UC3-SO-2	WP 2 (T2.2)
REQ-UC3-KM-2	The tenants should be able to generate,	<ul style="list-style-type: none"> - Confidentiality - Integrity 	Medium	REQ-UC3-KM-1	WP 2

	insert, retrieve and remove keys from their key management service	<ul style="list-style-type: none"> - Availability - Access by data owners - Upload/Download - Multi clouds 			(T2.2)
REQ-UC3-KM-3	The key management service should be able to offer different key types and generation algorithms to each tenant, e.g., AES128, AES256, 3DES etc.	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability 	Low	REQ-UC3-KM-1 REQ-UC3-DE-1 REQ-UC3-DE-3	WP 2 (T2.2)
REQ-UC3-KM-4	Only the tenants should be able to create and manage the keys	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Access by data owners 	High	REQ-UC3-KM-1	WP 2 (T2.3)
REQ-UC3-KM-5	The cloud service providers should have no access or visibility of the tenants' keys	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Access by data owners - Multi clouds 	High	REQ-UC3-KM-1	WP 2 (T2.3)
REQ-UC3-KM-6	The tenants should be able to cache their keys on trusted virtual machines or gateways in order to outsource or improve performance of the encryption and decryption process	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Selective sharing with other users/owners - Multi clouds 	Low	REQ-UC3-KM-1 REQ-UC3-DE-2	WP 3 (T3.1)
REQ-UC3-AC-1	Each tenant should be provisioned with	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability 	High	REQ-UC3-SO-1	WP 2 (T2.3)

	an instance of an access control service from the cloud service store	<ul style="list-style-type: none"> - Access by data owner - Multi clouds 		REQ-UC3-SO-6	
REQ-UC3-AC-2	The tenants should be able to create, delete and modify access control policies from their instance of the access control service	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Access by data owner 	Medium	REQ-UC3-AC-1	WP 2 (T2.3)
REQ-UC3-AC-3	The access control service should be able to offer use of different system and data attributes for the construction of a security rule, e.g., filesystem, user, application, and time attributes.	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability 	Low	REQ-UC3-AC-1	WP 2 (T2.3)
REQ-UC3-AC-4	Only the tenants should be able to create and manage their access control policies	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Access by data owner 	High	REQ-UC3-AC-1	WP 2 (T2.3)
REQ-UC3-AC-5	The cloud service providers should have no access or visibility of the tenants' access control policies	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Access by data owner 	High	REQ-UC3-AC-1	WP 2 (T2.3)
REQ-UC3-AC-6	All data protection operations should be governed by access control policies by either approving or denying access to	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability 	High	REQ-UC3-AC-1 REQ-UC3-KS-1	WP 2 (T2.3)

	the required keys				
REQ-UC3-AC-7	The access control service of tenants should be tightly coupled with their key management service, such that no key can be utilised without an approving access control policy	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Fine grained retrieval 	High	REQ-UC3-AC-1 REQ-UC3-KM-1	WP 2 (T2.3)
REQ-UC3-SO-1	Each tenant should be provisioned with a cloud service store account	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Access by data owner - Single cloud provider 	High		WP 4 (T4.3)
REQ-UC3-SO-2	The service store should provide the tenants with access to the storage services of multiple cloud service providers	<ul style="list-style-type: none"> - Multi clouds and federated clouds 	Medium		WP 4 (T4.3)
REQ-UC3-SO-3	The service store should be able to offer block storage service to the tenants	<ul style="list-style-type: none"> - Multi clouds and federated clouds 	High		WP 4 (T4.3)
REQ-UC3-SO-4	The service store should be able to offer object storage service to the tenants	<ul style="list-style-type: none"> - Multi clouds and federated clouds 	High		WP 4 (T4.3)
REQ-UC3-SO-5	The service store should be able to offer Big Data storage service (HDFS) to the	<ul style="list-style-type: none"> - Multi clouds and federated clouds 	High		WP 4 (T4.3)

	tenants				
REQ-UC3-SO-6	The tenants should be able to enable or disable the use of data protection service on the storage service of their choice	- Multi clouds and federated clouds	Medium		WP 4 (T4.3)
REQ-UC3-SO-7	The service store should be able to offer key management as a service to the tenants	- Multi clouds and federated clouds	High		WP 4 (T4.3)
REQ-UC3-SO-8	The service store should be able to offer access control as a service to the tenants	- Multi clouds and federated clouds	High		WP 4 (T4.3)
REQ-UC3-DE-1	The core encryption process should only be controlled and managed by the tenant	- Confidentiality - Integrity - Availability - Access by data owner	High		WP 2 (T2.1) WP 2 (T2.3)
REQ-UC3-DE-2	The tenant should be able to deploy and manage the core encryption process on trusted virtual machines or gateways as an agent or plug-in	- Confidentiality - Integrity - Availability - Selective sharing with other users/owners	Medium		WP 2 (T2.3) WP 3 (T3.1)
REQ-UC3-DE-3	The core encryption process should be FIPS 140 compliant	- Confidentiality - Integrity - Availability	Medium		WP 2 (T2.1)
REQ-UC3-DE-4	The encryption agent or plug-in	- Confidentiality - Integrity	Medium		WP 2

	should be able to access the tenant's key management service and access control service	- Availability - Selective sharing with other users/owners			(T2.3) WP 3 (T3.1)
REQ-UC3-DE-5	The keys should only be released to the encryption agent or plug-in upon approval of an access control policy	- Confidentiality - Integrity - Availability - Selective sharing with other users/owners	High		WP 2 (T2.3) WP 3 (T3.1)

5.4.7 Other Non-functional Requirements

Some non-functional requirements are:-

- **Configuration Management:** The deployment and configuration scripts should be exchanged between the service store and the VMs securely.
- **Transparency:** The encryption process should be transparent to the end user of the solution. The end-user should be able to use the block storage, object storage etc. in the protected mode in the same way as he uses them without any protection.
- **Concurrency:** The DPaaS solution should be multi-tenant and able to support requests originating from a multitude of concurrent users.

5.4.7.1 Performance Requirements

- **Life-cycle automation overhead:** The life-cycle automation for the provisioning process of the data protection service should carry a low-overhead of about 5% as compared to a standard provisioning process which does not include using the data protection service.
- **I/O performance overhead:** The encryption and decryption operations on the protected data should carry low I/O performance overheads, i.e., about 10-15% overhead for read operations and 20-30% overhead for write operations.

5.4.7.2 Compliance Requirements

- The core cryptography components of the data protection service should be FIPS 140-2 compliant.
- The data protection service should comply with the PCI DSS 3.0 requirements 3, 7, 8, and 10.

- No customer data should be lost when performing encryption or decryption operations or as a result of the provisioning or de-provisioning of the data protection service.

5.4.7.3 Software Quality Attributes

- Interoperability: The key management as a service feature should support the OASIS Key Management Interoperability Protocol (KMIP) and the Public-Key Cryptography Standard 11 (PKCS #11)
- Maintainability: The key management as a service feature should support key expiration and recovery operations

5.4.7.4 Business Rules

- The data owners / tenants are able to subscribe to the data protection service and protect their data on the cloud platforms.
- The data owners / tenants are able to enable or disable their use of the data protection service up to the level of virtual machines.
- The end-users are able to install the encryption agents and plugins on virtual machines and gateways if they have the administrative rights on them.
- The service store admin is able to integrate, modify and remove the provisioning of storage services of different cloud providers for the data owners.

5.4.7.5 Other Requirements

- The core encryption procedures, as well as access control and key management practices developed in this use case should not contradict the EU data protection laws.
- The data protection system should be configurable and modular enough to allow localisation either by specifying the localisation constraints in the policies or by replicating or isolating key stores.

6 Elaboration of USE CASE 4 with Security Requirements

6.1 Introduction

Problem statement: Cloud services are designed to provide easy, scalable access to applications, resources and services and are fully managed by a cloud services provider.

Currently the communication between a customer and a CSP is encrypted. Also, the client needs credentials to access the service. These provisions may seem sufficient; however some security problems still exist in CSPs:

- The database management systems (DBMS) operate on clear text, so files are not protected against a malicious user that gains access to the System Server.
- The administrator of the Cloud Service can access the customer data.

Contribution: ESCUDO-CLOUD aims to solve these problems: It will encrypt all data files transmitted from client device (mobile or computer) to the CSP. The data files will remain encrypted in the cloud. The user will have to use their credentials (or other security mechanisms) to access their records. Access is achieved through a web browser or an agent that is installed on the user's computer or mobile device.

This use case outlines a first approximation of the actors and requirements (functional and non-functional) that will exist in the ESCUDO-CLOUD project. The objective of ESCUDO-CLOUD is to provide security to the Cloud Service Providers. ESCUDO-CLOUD will act between the customer and CSPs to fulfil its purpose.

6.2 High Level “Business” Use Case

6.2.1 Introductory Scenario

Cloud computing offers myriad benefits to customers: scalable storage, applications improved collaboration regardless of team members' locations, etc. We could say that cloud computing is to the 21st century what electricity was to the 20th: it's revolutionizing the way we do business. Furthermore, cloud computing is a rapidly growing business: IDC predicts [7] that the public cloud market will grow by 23% per year, over the next five years (from around \$48 billion today to \$130 billion by 2018).

However there is a major concern when it comes to cloud computing: how safe is the cloud? There can be downsides and potential risks when relying on a third party to provide infrastructure (IaaS), platforms (PaaS) or software (SaaS). In our case, providing storage can cause distrust to customers as they are relying on the provider to protect their data. For instance, the software that manages the cloud servers may have security vulnerabilities. Therefore customer's data can be lost, stolen or damaged. Whether changes are malicious or accidental, when multiple tenants share a public cloud infrastructure, it can be challenging to audit the integrity of your cloud-stored data.

This business use case shows how a Single Cloud Service Provider (CSPs) can benefit from ESCUDO-CLOUD. By using ESCUDO-CLOUD CSPs are able to store encrypted information from the client side. Data owners will give more trust to the CSPs and use their services for a wider range of applications, possibly moving most of their resources to the cloud. CSPs will significantly benefit too due to the increased market penetration that robust data ownership would provide. There are also secondary benefits to the CSP in the form of reduced regulatory risks, audit costs, and general security threats that they would typically have to face in the absence of such protection. Figure 25 shows a simple scenario with ESCUDO-CLOUD:

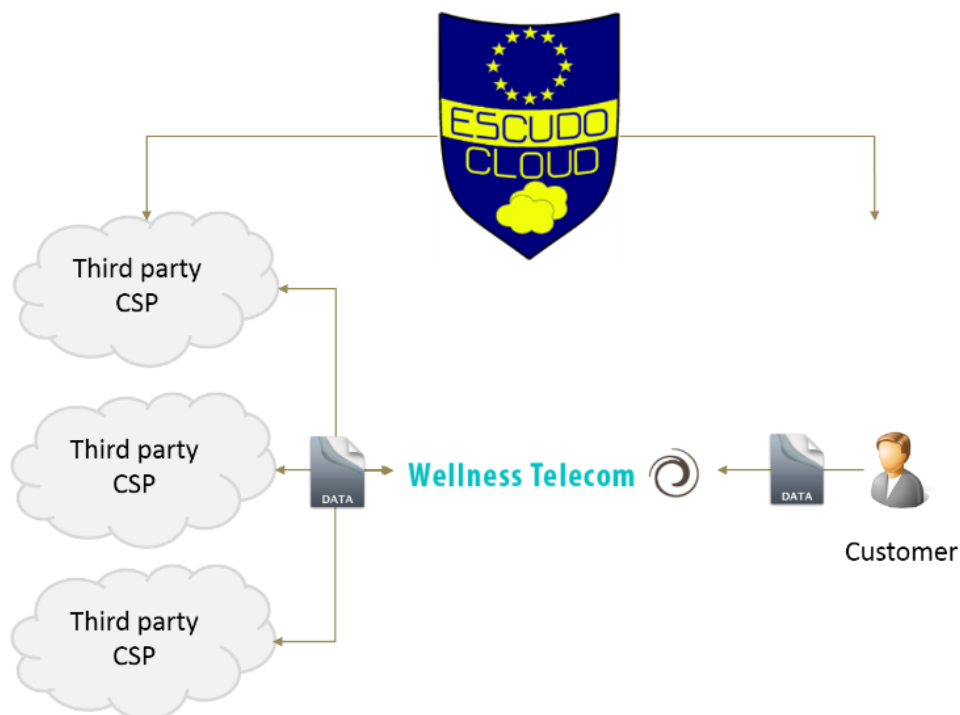


Figure 25. Introductory scenario for Use Case 4

Freeing providers from the worries of protecting data, allows them to handle the data outside of the control of their internal IT infrastructure. For instance, it would enable a provider itself to rely on other services for outsourcing storage and computation, behaving as a broker providing a virtualised cloud service, without worrying about the possible improper exposure of user information, which is guaranteed to be self-protected. By using ESCUDO-CLOUD customers can trust that the CSPs will correctly manage the outsourced information.

6.2.2 Purpose

The “Elastic Cloud Service Provider” Use Case of the ESCUDO-CLOUD project will allow companies to become a “virtual” CSP, leveraging computational and storage infrastructures (i.e., IaaS) by third providers, and to secure its services offering certifiable secure data management in the cloud.

Cloud services offer ease and convenience – and opportunities for malicious actors to steal data or worse, keys. Encryption is a vital component of any organization’s security strategy. Trust in the cloud is based on a full accounting of where your organization’s data needs to be secured and at what level. Proper encryption that enables you to retain control of the keys will protect your data and give you confidence in the cloud.

ESCUDO-CLOUD will be able to offer the option to replicate in a secure way data across several CSPs, the availability of the data will remain unaffected despite outages of individual clouds. Thanks to this technology CSPs will be able to increase its business and would tend to unlock the market barriers that are a challenge to CSPs with limited resources and a smaller foothold in the cloud storage market. ESCUDO-CLOUD will enable companies to undertake new joint ventures with other small CSPs or even with large players to be able to offer secure services to their customers all around Europe.

6.2.3 Comparison to current practice

Current state of cloud services operates on sensitive data almost constantly. Though data at rest may be stored encrypted (and therefore securely), when it is operated on, it must be decrypted. This constant encryption and decryption of user data means that the decryption key is present in RAM somewhere. It may be in the OS, it may be in the Database Management System (DBMS), or it may even be in the application itself.

This means that a user with access to a system inside the CSP can access the database decryption key, or potentially even the unencrypted database contents, from the RAM, or ‘working memory,’ of the computer. As a result, the robustness of the database encryption scheme becomes nearly irrelevant and would likely not have posed a substantial barrier to someone with the capability to circumvent authentication protocols in the first place.

Some providers combine powerful data encryption with patented holomorphic split-key encryption technology to increase security and protect keys. One part of the key is hosted by the provider, while the second part – the master key – is held by the customer. The result is that customers control their data and do not need to trust anyone else with their keys.

This technology easily encrypts any disk or data storage unit with proven encryption algorithms such as AES-256 and makes it safe from hackers, unauthorized access, competitors, and other threats.

ESCUDO-CLOUD not only encrypts all data files stored on the CSP, but also offers a uniform user interface (namely a web portal or agent), irrespective of what CSP is being used. Thanks to this middleware the user will be able to access to the data files stored in the cloud.

6.2.4 Objectives

The objective of ESCUDO-CLOUD is to offer certifiable secure data management in the cloud. Some of its main features are:

- Encrypting the communication between the client and the CSP.
- Encrypting the files stored in the cloud.
- Establishing and defining clear roles.
- Allowing access file through a web browser or agent.

6.2.5 Stakeholders

There are three stakeholders involved in the system. Each stakeholder will be related to a different entity:

- End User: This can be the data owner itself or another user with access rights to shared files. The different user roles are elaborated on in Section 3.1. Each end user will interact with ESCUDO-CLOUD to access the files that are stored in the CSP. This interaction can be achieved via their computer (as if they were located in a local folder), from a mobile device (iOS, Android) and through a web interface (HTTPS).
- Broker: the stakeholder identified with this profile will be WT. His goal is to offer a user interface. The customer will be able to access his file through this interface. The broker can offer an interface like a web portal, an application, etc. Although the stakeholder is defined, it can take several forms.
- CSP: the role of CSP may be developed for WT, but using an elastic CSP may be necessary. In this case, Amazon or any other cloud service provider will be used. Its aim is to store the data file of several users. This information will be encrypted in the cloud. Only its owner will be able to access to it.

These stakeholders are clearly defined in all possible scenarios. ESCUDO-CLOUD will act to secure the data files in the CSP.

6.2.6 Glossary of Acronyms

Acronym	Definition
UC	Use Case

CSP	Cloud Service Provider
LDAP	Lightweight Directory Access Protocol
IaaS	Infrastructure as a Service
SaaS	Software as a Service
PaaS	Platform as a Service
DBMS	DataBase Management Systems
HTTPS	HyperText Transfer Protocol Secure
SSO	Single Sign-On

6.3 Use Cases

The use case that will be explained in this section describes how a final user act with ESCUDO-CLOUD to access to his data files.

6.3.1 Overview

This use case diagram describes the relationships between the customer and the features provided by ESCUDO-CLOUD. End Users will be able to access data with different levels of access (read/write permissions) from their computer (as if they were located in a local folder), from a mobile device (iOS, Android) and through a web interface (HTTPS). Furthermore, if the user owns third-party storage services such as Dropbox or Google Drive, they can be included in this platform. The Figure 26 shows the Use Case diagram:

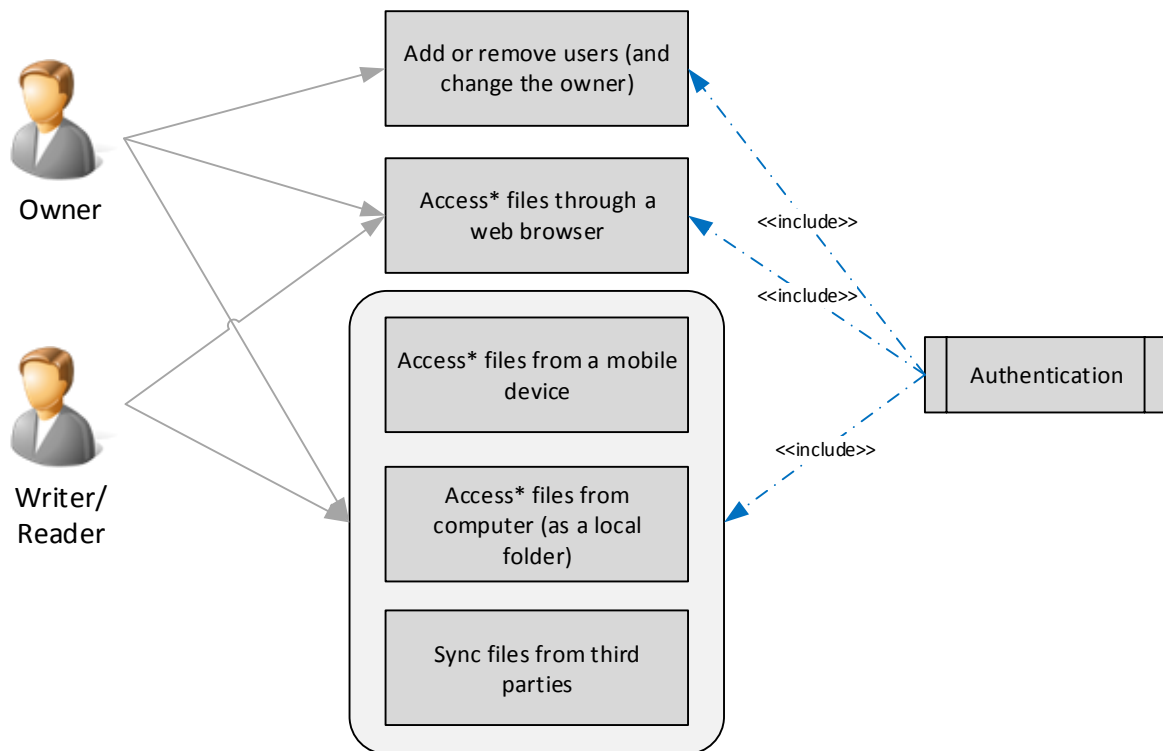


Figure 26. Use Case Diagram (Access* means the privileges of each role of user)

Each End User registered in the system purchases some disk space (they can buy more space if needed) and within this, they can create a Share. This is a special folder, the contents of which can be shared with other End Users of the system.

There will be three types of End User with the following privileges:

- **Owner:** This user (unique for a Share) has full access to the data (read, write, modify and delete) and can also add or remove users (knowing their identifier) to the Reader or Writer groups; the Owner can pass the ownership of a Share to another End User. In this case he will lose his owner privileges and become a Writer - the selected End User will become the new Owner of the Share. When an Owner wants to add an End User to one of the groups a request will appear on the interface of the selected End User which can be accepted or rejected.
- **Writer:** Users belonging to this group can read, write, modify and delete files present in the Share
- **Reader:** Users belonging to this group can only read the files present in the Share

Each user registered in the system can create a Share and become the Owner of it.

Each individual Use Case is described below:

- **Add or remove users (and change the owner)**

The purpose of this use case is to let the owner of the file add or remove new users in the file. Those users will be able to read or write the file (it depends of the privileges that the owner gives them). The owner will be able to assign a new owner for the file (and then he will lose his role).

- Access files through a web browser

The purpose of this use case is to let the end user access the files through a web browser, via web portal with his credentials. Access can be *only read* or *read, write, modify and delete* (it depends on the role of the user).

- Access files from a mobile device

The purpose of this use case is to let the end user access the files using a mobile device (with an agent previously installed). Access can be *only read* or *read, write, modify and delete* (it depends on the role of the user).

- Access files from computer (as a local folder)

The purpose of this use case is to let the end user access the files from a computer. He will need an agent previously installed too, and then will be able to access his files through a local folder. The credentials that the user uses to login in the computer will be used to login in ESCUDO-CLOUD, so he won't need introduce his credentials again. Access can be *only read* or *read, write, modify and delete* (it depends on the role of the user).

- Sync files from third parties

The purpose of this use case is to let the end user sync files that he has stored in other third-party services (like Dropbox or Google Drive).

The general architecture of the use case 4 is shown in the Figure 27. The user can access the data files stored in the cloud using an agent or through a web portal. The operation is similar in both cases:

1. The user inputs the credentials.
2. The ESCUDO-CLOUD middleware (agent or web portal) validates the credentials.
3. The user can manage the data files stored in the cloud through the middleware. The information will be encrypted and decrypted on the client side in both cases.

The only difference is that an internal communication will exist between the agent and the WT web portal when a user access through this via, but this communication will be transparent for the data owner.

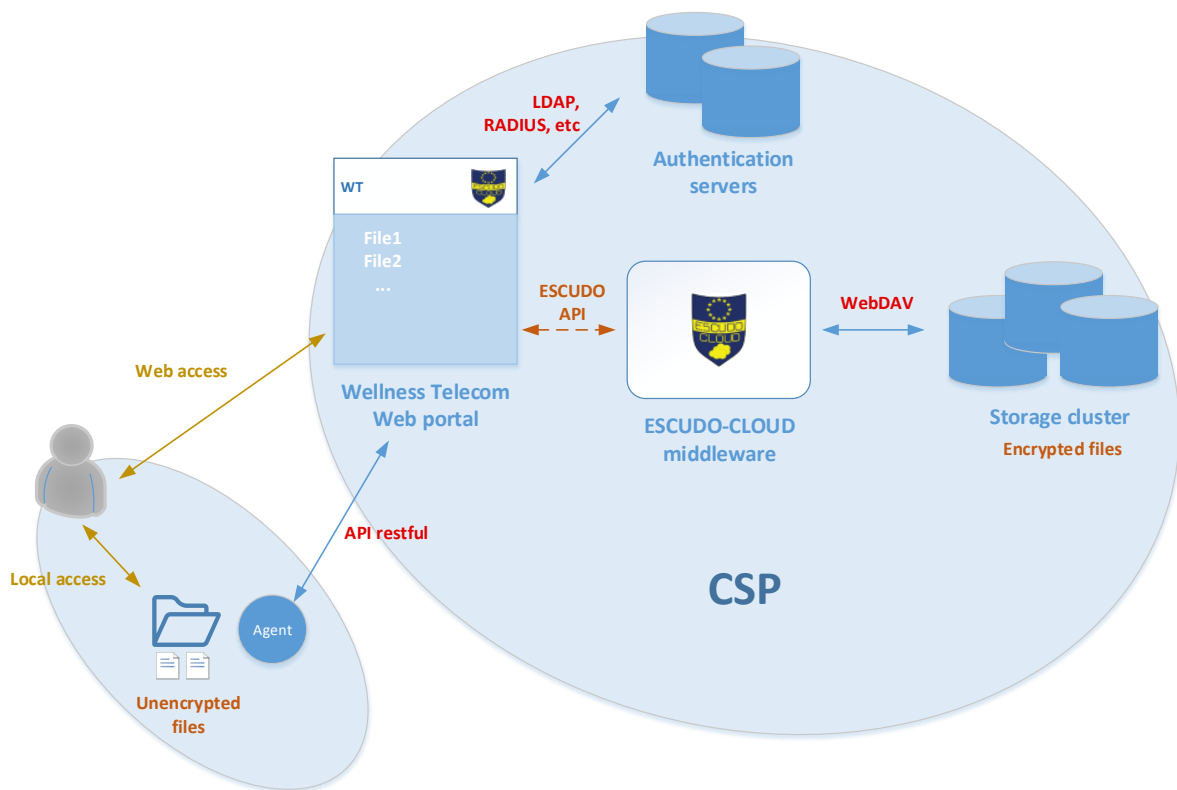


Figure 27. UC4 – General architecture

6.3.2 Use Case: Elastic Cloud Service Provider

As the previous diagram showed, the End User can access the system through a web browser or agent (mobile device, local folder...). The sections below describe the characteristics about this use case.

6.3.2.1 Actors

- End Users
- Broker (service web portal or agent)
- CSP (storage)

6.3.2.2 Purpose

The main goal of this use case is to have access to all data from anywhere through a web browser. Moreover, it is possible to access all data with an agent - software installed on the computer or a local folder.

6.3.2.3 Priority

- Data must be encrypted at rest when stored with the CSP

- On the client side, a user must be able to access files stored on the CSP for which they have permission (i.e., they are the Owner of the data or a member of the Writer/Reader group for the Share)
- Data transfers between the client and the CSP must be encrypted to provide protection for the data in transit.
- User and Password credentials must be used at a minimum to provide the authentication controls to ESCUDO-CLOUD (and therefore the data on the CSP). More sophisticated authentication mechanisms such as certificates can also be used.

6.3.2.4 Pre-conditions

- In addition to the use of encryption to protect the user data there should be an authentication process based on credentials (user and password), certificates, Multi-factor authentication (MFA), etc. to allow users log into the web platform.
- The agent must be installed, unless the end user has access to ESCUDO-CLOUD through a web browser.

6.3.2.5 Sequence of events

Figure 28 shows the main events that occur during communication between a data owner and the CSP (where his files will be encrypted). The interaction with the CSP through ESCUDO-CLOUD should be transparent so that to the user it appears that they are working directly with the CSP.

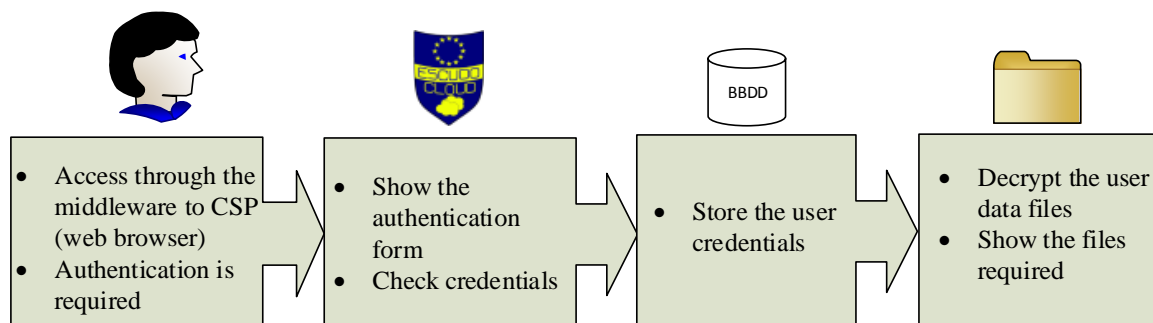


Figure 28: Sequence of events with web portal

If the user accesses the cloud using an agent (for instant an application installed in his device), the user will provide his username and password on the computer or mobile device, and with these credentials will access the ESCUDO-CLOUD agent directly, so it is a Single Sign-on (SSO) service.

6.3.2.6 Practical implementation

All files will remain encrypted on the server side. There will be a mechanism to map the real and shadow file names.

When End Users enter their credentials in the web form, ESCUDO-CLOUD middleware will be able to identify and decrypt the files requested by the user. It doesn't matter if the user accesses through the web browser or the agent, the performance is similar in both cases, as shown in Figure 29 and Figure 30.

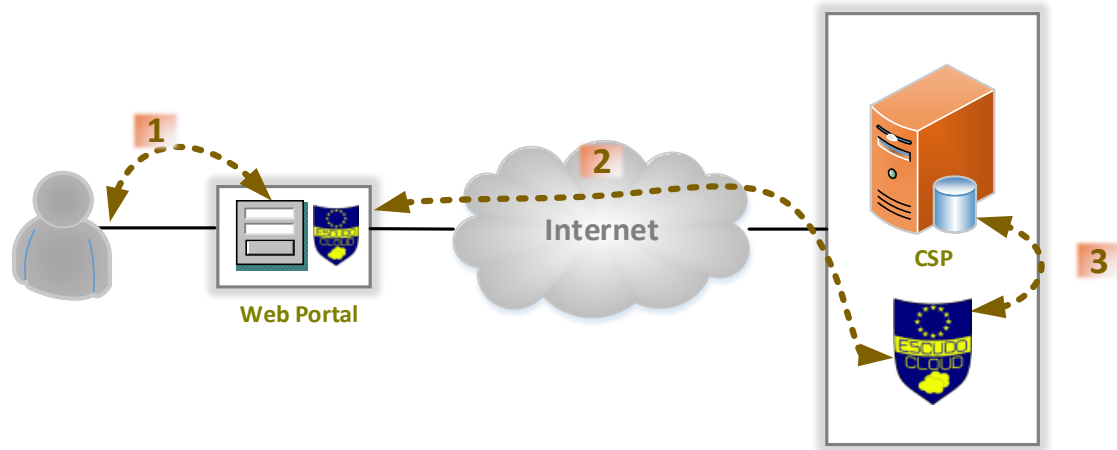


Figure 29. Access through a web browser

The main difference between access with and without agent is how the internal communication is done, but the whole process is transparent for the user in both cases: the end user will access his files through his web browser or agent (like a folder for instance).

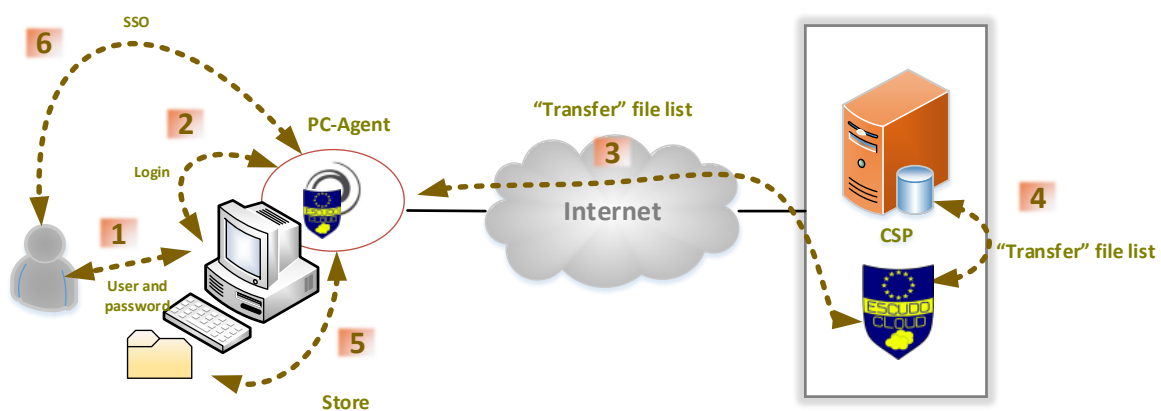


Figure 30. Access through an agent

6.3.2.7 Post-conditions

All data will remain encrypted on the CSP for future access request although the user ends the session.

6.3.2.8 Alternatives

Instead of username and password credentials, users may authenticate via certificates or by using a Lightweight Directory Access Protocol (LDAP).

6.3.2.9 Exceptions

- ESCUDO-CLOUD can't connect to the CSP: it won't be possible synchronize data files and the user only will be able to access to the stored files in the agent.
- There are latency issues between ESCUDO-CLOUD and the CSP or ESCUDO-CLOUD and the client: the middleware will notify the user.

6.3.2.10 Storyboard

For this use case, the storyboards are mock-ups that show the main actions between the user and the interface. These mockups are shown in Figure 31 to Figure 35:

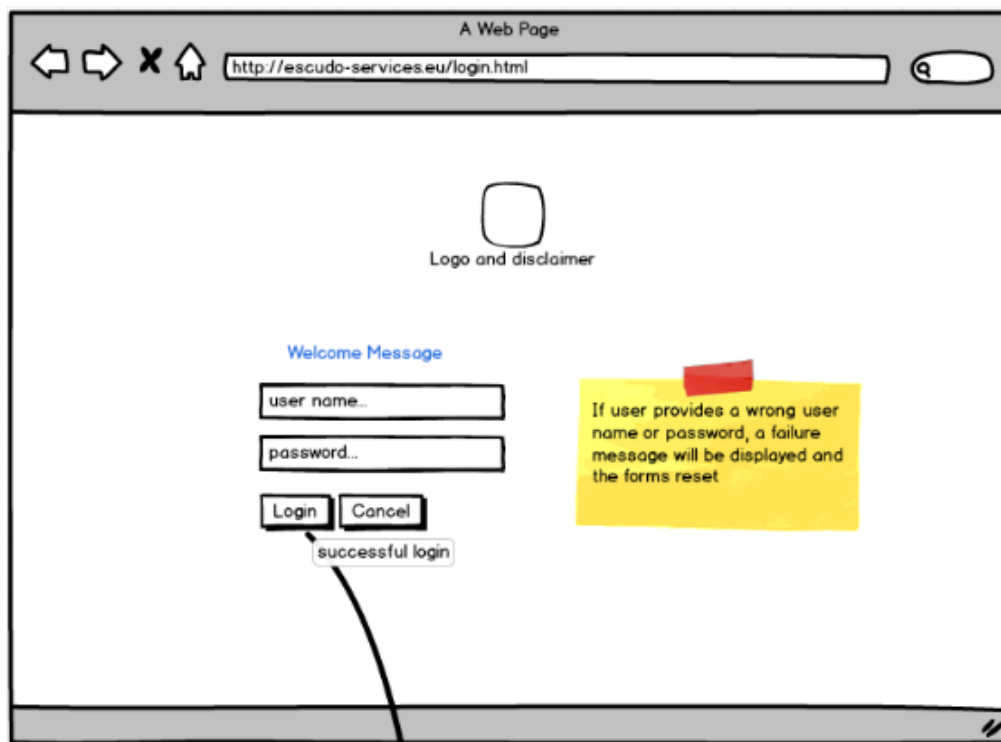


Figure 31. ESCUDO-CLOUD portal in web browser, user and pwd prompt

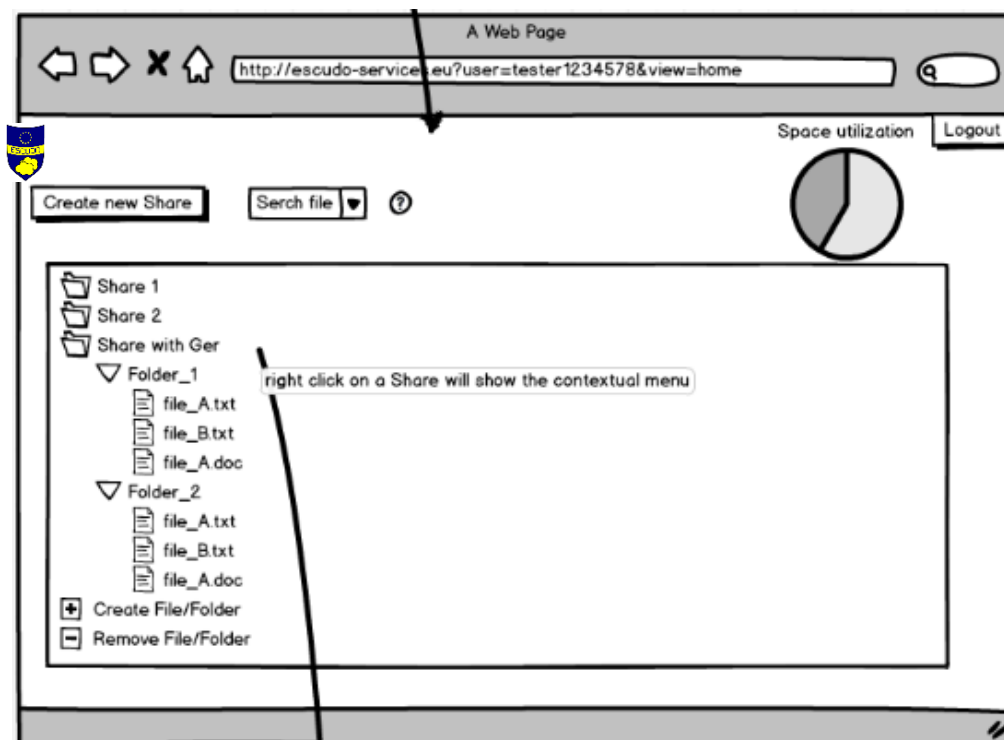


Figure 32. Page for user listing files and folders accessible

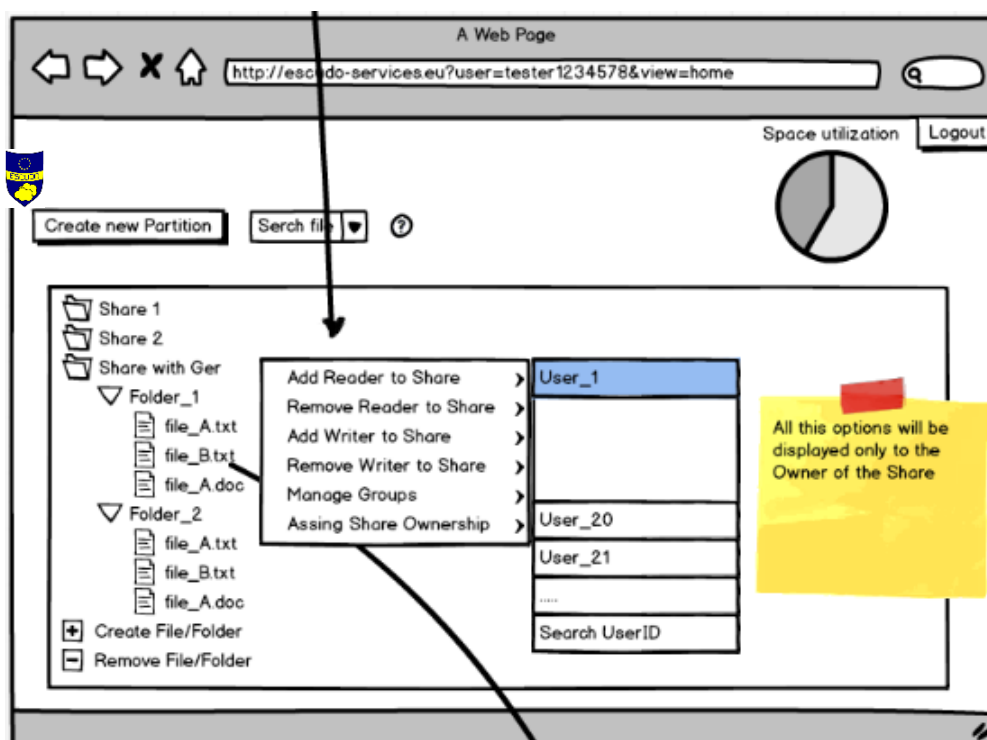


Figure 33. User can consult and change the privileges of his files

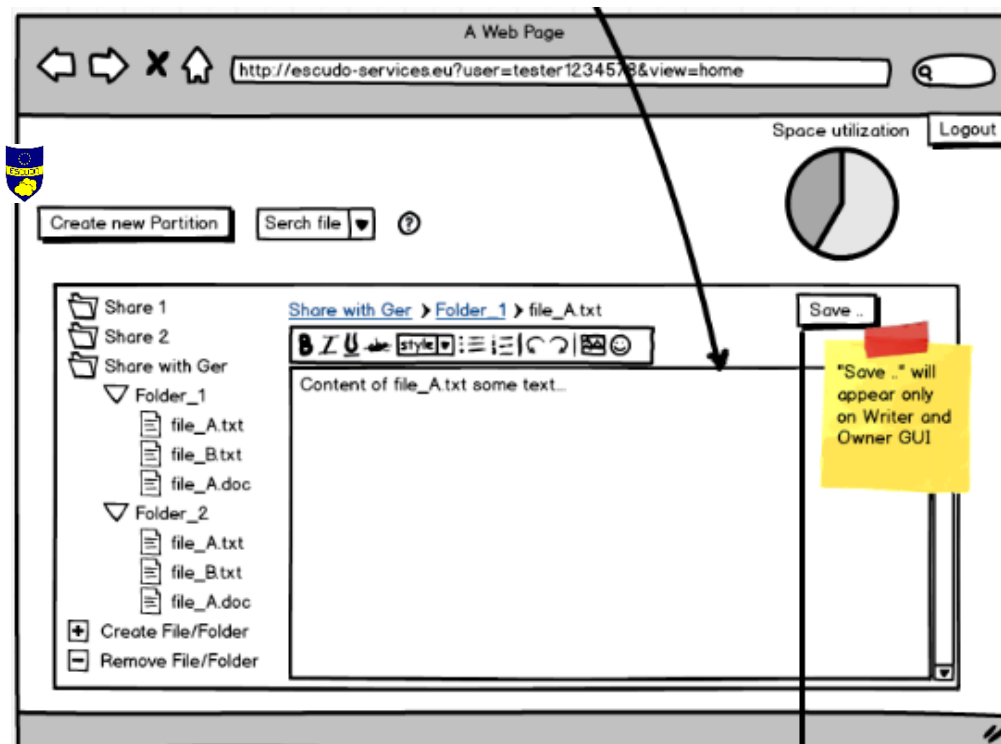


Figure 34. User belong to Writer group or Owner successful opens a file

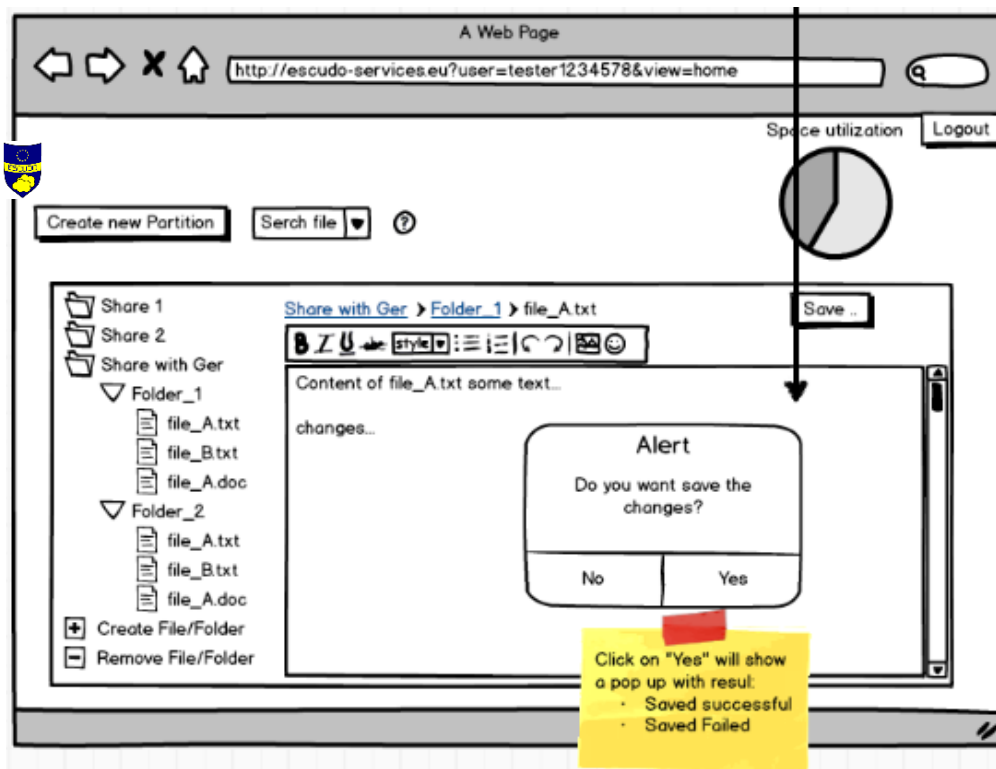


Figure 35. User belong to Writer group or Owner click on “Save” button

6.4 Requirements for Elastic Cloud Service Provider

6.4.1 Introduction

6.4.1.1 Purpose

The scope of this solution is to offer a middleware to permit the user access to his stored data in his CSP. For this purpose, the end user will be able to use an agent or access through a web browser.

6.4.1.2 Conventions

None.

6.4.1.3 Major relevant ESCUDO-CLOUD dimensions

The ESCUDO-CLOUD middleware has several dimensions that allow the users to manage their stored files safely and easily. To fulfill this purpose, ESCUDO-CLOUD has the dimensions shown in the Table 1.

This use case “Elastic Cloud Service Provider” operates in all of these four dimensions. The points below show a short definition about their scope:

- Security

The middleware of this use case must ensure the “CIA”: confidentiality, integrity and availability for the end user and his data files. To that end, the files will remain encrypted on the CSP and only the correct user that accesses through the middleware will be able to decrypt the information.

- Sharing

An end user will be able to share a data file in the cloud depending on the role which he access to the middleware (via web browser or agent). It will exist three different roles on ESCUDO-CLOUD described below the owner user, the writer user and the reader user and not all of them will have the same privileges to manage the data files stored in the cloud.

- Access

For our use case, the access dimension will describe the conditions that the middleware must be in compliance to permit the user access to his data stored in the cloud through different ways.

- Cloud Architectures

The ESCUDO-CLOUD middleware can work with different cloud architectures, namely: single, multi or elastic cloud. The middleware must permit transparent access to the end users no matter the type of cloud that is operating behind.

6.4.2 Initial solution architecture in the context of the use-cases

The solution permits the user access to the cloud where they have their data files through a common interface. They will be able to use a web browser or a device with an agent to manage all their information stored in the CSP.

The main benefit to the users is the uniformity and transparency that the middleware gives them. Also, with ESCUDO-CLOUD middleware they have all their files encrypted in the cloud and only the users with the correct privileges will be able to decrypt (through the middleware) the information.

6.4.3 Overall Description

6.4.3.1 Solution Perspective

The architecture for the Access files through a web browser or agent use case is shown in the Figure 36:

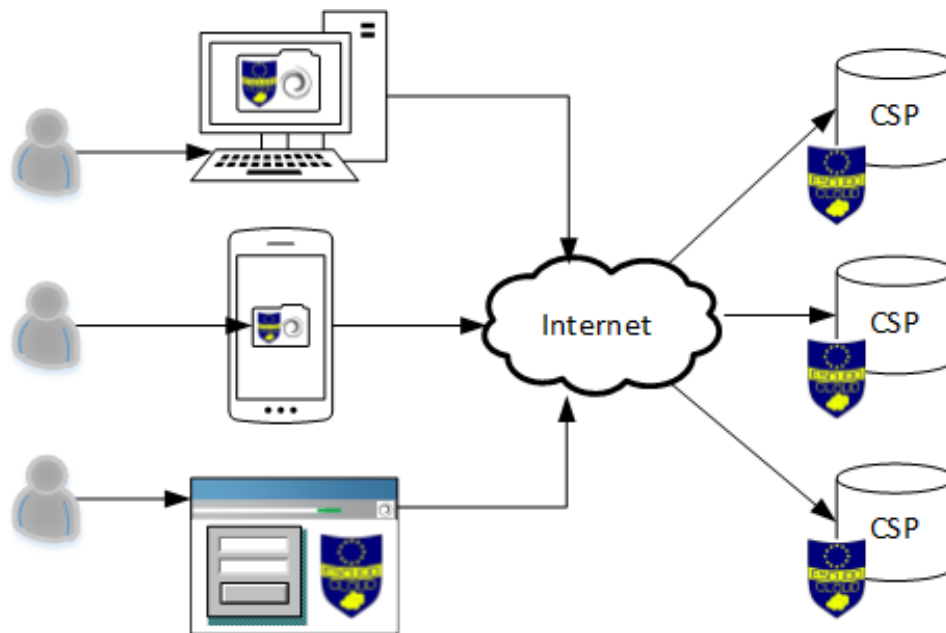


Figure 36. General Architecture for the access

The middleware permits the user access to the CSP in different ways. For instance, they can use a web browser in their computers or a local folder. Today, the use of mobile devices is as important as the use of a computer, so it is possible install an agent on the mobile device too. In all of them there will exist an instance of ESCUDO-CLOUD. The login will be different depending on the type of access.

On the other hand, on CSP side, it is necessary another instance of ESCUDO-CLOUD to make possible the communication between the user side and the cloud side. Depending on the type of

access (with or without agent) the files will be decrypted on a different side. But, in general, the data files will be encrypted in the cloud. Not even the administrator of the CSP will be able to access to the clear information.

With the middleware, all these processes of communication are completely transparent to the end user; this is one of the most important objectives of this use case.

6.4.3.2 Solution Functions

- Uniform access for the user through the middleware
- Safe storage in the elastic cloud service provider

6.4.3.3 Actors and Characteristics

The following points show the actor that will be used in this use case:

- CSP

The CSP will use several types of clouds, but WT proposes an elastic cloud solution due to its flexibility to change its size depending on the requirement of each moment. Its aim is to store the encrypted data files of multiple end users.

- Broker

It is the middleware that permits the communication between the end users and the CSP. It can take several forms: web portal, local folder, application... But its objective is clear: it offers a user interface to the CSP.

- End user

There will exist different types of roles of end users (owner, writer or reader) and their function is to access their data files stored in the cloud (according to their privileges). They will use the broker for this purpose.

6.4.3.4 Operating Environment

The platform of the Wellness Telecom Cloud, devoted to virtualization is based on: VMWare vSphere 5.5. This solution brings features that are transparent to the client like load balancing, high availability of the hardware that underlies the solution, etc.

6.4.3.5 Design and Implementation Constraints

There are some design and implementation constraints in order to work with an elastic cloud.

- Depending on the elastic cloud service, the users will be able to send a limited number of instances in a period of time.
- In order to use the agent to access to the stored files in the cloud, the agent will have to be installed previously.

6.4.3.6 Assumptions and Dependencies

There are several assumptions and dependencies for the components of the use case “Access files through web browser or agent”. The following points show these features:

- The middleware offers a user interface to access CSP in a transparent manner.
- The end user will have a login to identify himself in the system.
- The administrator of the CSP can manage the stored service (for instance, using a third party cloud) but cannot access the clear information of the user files.
- In the agent access scenario, end user should install the software agent previously.
- The communication between the ESCUDO-CLOUD agent of the user side and the ESCUDO-CLOUD agent of the CSP side is transparent to the end user.
- The broker is the responsible of the decryption of the data files, as on the CSP side as on the user side.

6.4.4 External Interface Requirements

6.4.4.1 User Interfaces

The user interfaces will change depending on the type of access: through a web browser or an agent. If the user accesses the CSP using the agent, he will not have to use his credentials, it is sufficient the login on the computer or device. However, all end users will have to introduce their credentials every time they access through a web browser. So, we have two groups of user interfaces:

- **CSP**
 - A login interface for the CSP administrator.
 - A dashboard interface for the CSP administrator to modify the cloud features.
- **Broker**
 - Web browser interface
 - A login interface for the end user.
 - A dashboard interface for the end user to view, add, delete or modify his data files (depending on their credentials).
 - A login interface for the web administrator.
 - A dashboard interface for the web administrator to make changes on the web portal.
 - A dashboard interface for the creation of new accounts.
 - Agent interface

- A dashboard interface for the end user to view, add or delete his data files (depending on their role).
- A login interface for the administrator of the agent.
- A dashboard interface for the administrator of the agent to make changes or updates.

6.4.4.2 Hardware Interfaces

The solution will be hardware independent.

6.4.4.3 Software Interfaces

The following software interfaces are required in the system:

- APIs for the communication between ESCUDO-CLOUD middleware on the user side and the ESCUDO-CLOUD middleware on the CSP side.
- APIs for the communication between the CSP and the third parties of the elastic cloud.
- APIs for the installation and operation of the agent on the operative system. The agent should be compatible with both mobile operating systems (Android, iOS, Windows, etc.) and computer operating systems (Windows, iOS, UNIX, etc.).
- API between the agent or web portal and the ESCUDO-CLOUD decrypter on the client side.

6.4.4.4 Communications Interfaces

The following communications interfaces are required in the system:

- APIs for the operation of the web portal on the web browser. The web portal should be compatible with the following web browsers: Google Chrome, Mozilla Firefox, Internet Explorer, Safari, etc.
- The login interface and the dashboard for the user has already been shown on the storyboard section.

6.4.5 System Features

6.4.5.1 Access Control (AC)

6.4.5.1.1 Description and Priority

Description	The access control is the feature which permits the user to connect with the CSP through the ESCUDO-CLOUD broker. Thanks to the access control each user will access to the system with a group of privileges.
Priority	High

Risk	9 (wrong privileges)
-------------	----------------------

6.4.5.1.2 Stimulus/Response Sequences

It is necessary that the user enter their username and password to use the service of ESCUDO-CLOUD.

An end user will be able to manage, one way or another, the data files stored in the cloud. When the user access to the ESCUDO-CLOUD middleware, he will be classify in one of the following roles for each data file:

- **Reader** will be able to only read files.
- **Writer** will be able to read, modify and delete files.
- **Owner** will be able to read, modify, delete and change the owner of the files.

6.4.5.1.3 Functional Requirements

REQ-UC4-AC-1: Each user should have a unique username and password to enter in the system through a web portal. The user will login in the middleware using his credentials, and only with them will be able to access to his stored data files.

REQ-UC4-AC-2: by using the agent, it should be possible to remember the credentials. When a user accesses for the first time the agent installed in his device (computer, mobile...) he will login with his username and password, but he should not need to introduce them again.

REQ-UC4-AC-3: each pair username/password should be associated with a specific role. The credentials are the form to distinguish what privileges has each user. For example, a reader user should not be able to modify a data file.

REQ-UC4-AC-4: only the file owner should be able to authorize another user for reading or modifying his data files, that is to say that the file owner will assign the role of the rest of the users.

REQ-UC4-AC-5: only the administrator of the system (the ESCUDO-CLOUD middleware) should be able to enable the access of a locked user. This is a prerequisite for the REQ-UC4-AC-6.

REQ-UC4-AC-6: each user should have a limit of attempts to access the system. For example, if a user introduces five wrong credentials, he will be locked.

6.4.5.1.4 Relevant ESCUDO-CLOUD project dimensions

The relevant ESCUDO-CLOUD dimensions are described below:

- Security

The username and the password classify the user in roles with different privileges, so the data files are protected from actions of the rest of the users. Limiting the attempts to access is a first protection from non-allowed users.

- Sharing

An end user will be able to share a data file in the cloud depending on the role which he access to the middleware (via web browser or agent). There will exist three different roles on ESCUDO-CLOUD described below the owner user, the writer user and the reader user and not all of them will have the same privileges to manage the data files stored in the cloud. Depending on the credentials of the users, they will have a role or another.

- Access

The pair username/password allows that an end user access to his stored data files (or non-owner authorized data files), via web portal or with an agent.

- Cloud Architecture

It is not a relevant dimension on this feature.

6.4.5.2 Storage Service

6.4.5.2.1 Description and Priority

Description	The storage service is offered by the cloud. It is necessary an adequate the storage capacity.
Priority	Medium
Risk	7

6.4.5.2.2 Stimulus/Response Sequences

The administrator of the cloud, in our use case WT, will manage the features of the cloud. The users will be able to access to the storage service, with their correct privileges, through the ESCUDO-CLOUD middleware.

6.4.5.2.3 Functional Requirements

REQ-UC4-SS-1: the cloud (or storage service) should have capacity enough to store the data files of the users.

REQ-UC4-SS-2: the storage services should be accessible for the end users through ESCUDO-CLOUD middleware.

REQ-UC4-SS-3: by using an elastic cloud should be possible to adapt the capacity of the cloud to the user requirements. It is a pre-requisite for the requisite REQ-UC4-SS-1.

REQ-UC4-SS-4: the storage service should comply with “EU Directive 95/46/EC – The Data Protection Directive”. The personal data is defined under Article 2 (a): “*personal data shall mean any information relating to an identified or identifiable natural person ("data subject"); an*

identifiable person is one who can be identified, directly or indirectly, in particular by reference to an identification number or to one or more factors specific to his physical, physiological, mental, economic, cultural or social identity". This is a pre-requisite for the requisite REQ-UC4-SS-5.

REQ-UC4-SS-5: the CSP should implement appropriate technical and organizational measures to protect personal data against accidental or unlawful destruction or accidental loss, alteration, unauthorized disclosure or access, in particular where the processing involves the transmission of data over a network, and against all other unlawful forms of processing. This is a pre-requisite for the Data Encryption requirements below.

6.4.5.2.4 Relevant ESCUDO-CLOUD project dimensions

The relevant ESCUDO-CLOUD dimensions are described below:

- Security

The storage service grants confidentiality, integrity and availability of the stored data files. It will implement appropriate mechanism to fulfil this objective.

- Sharing

An end user will be able to share a data file in the cloud depending on the role which he access to the middleware (via web browser or agent). The cloud will have capacity enough to store these data files.

- Access

The access dimension will describe the conditions that the middleware must be in compliance to permit the user access to his data stored in the cloud through different ways

- Cloud Architecture

Elastic cloud architecture will allow the store service adjust his size to the user requirements and, by using the ESCUDO-CLOUD middleware, the end users will be able to access to the cloud regardless of the internal architecture of it.

6.4.5.3 Data Encryption

6.4.5.3.1 Description and Priority

Description	The data files will be encrypted on the CSP side. The decryption will be done on the client side.
Priority	High
Risk	9

6.4.5.3.2 Stimulus/Response Sequences



There exist two different points of view for this feature, but in both cases the user will access his data files to read, modify, etc.:

a) A user accesses the web portal to manage his stored data files

In this case, the decrypter will take action on the CSP side because the files will be shown to the end user through a web browser.

b) A user accesses an agent to manage his stored data files

In this case, the decrypter will take action on the user side. To use the agent, the files have to be synchronized previously, so it is necessary that they go through an insecure network. The files will travel encrypted and they will be decrypted on the end user side.

6.4.5.3.3 Functional Requirements

REQ-UC4-DE-1: The files should be stored encrypted. Only the user with the correct privileges will be able to decrypt the information, not even the administrator of the server will be able to access the content of the user data files. This is a pre-requisite for REQ-UC4-DE-2 and REQ-UC4-DE-3.

REQ-UC4-DE-2: The data files should remain encrypted on the CSP until the user accesses them through a web browser.

REQ-UC4-DE-3: the data files should remain encrypted on the CSP until the user synchronizes the server with the agent installed on his device.

REQ-UC4-DE-4: an end user should be able to securely download a data file from the CSP through the web browser. To fulfil this purpose, the CSP should have enough capacity to make an “on flight decryption”.

6.4.5.3.4 Relevant ESCUDO-CLOUD project dimensions

The relevant ESCUDO-CLOUD dimensions are described below:

- Security

By using encryption mechanisms, it is possible to obtain confidentiality and integrity on the system.

- Sharing

The users can share (upload and download) the data files safely.

- Access

The encryption and decryption will be performed on the the client side, in order to ensure the communication and not compromise the safety of the information.

- Cloud Architecture

For this feature, the dimension of cloud architecture is important because the CSP has to have capacity enough to support “on flight decryption”.

6.4.6 Requirements Catalogue

Table 5. Requirements Catalogue for Use Case 4

Requirement Reference #	Requirement Description	ESCUDO-CLOUD dimension	Priority	Dependencies on other requirements	Relevant Core Work Package
REQ-UC4-AC-1	Each user should have a unique username and password to enter in the system through a web portal.	<ul style="list-style-type: none"> - Confidentiality - Upload/download - Fine-grained retrieval - Write operations - Access by data owners 	High		WP2 (T2.3)
REQ-UC4-AC-2	By using the agent, it should be possible to remember the credentials.	<ul style="list-style-type: none"> - Confidentiality - Upload/download - Fine-grained retrieval - Write operations - Access by data owners 	High		WP2 (T2.3)
REQ-UC4-AC-3	Each pair username/pass should be associated with a specific role.	<ul style="list-style-type: none"> - Confidentiality - Upload/download - Fine-grained retrieval - Write operations - Access by data owners 	High		WP2 (T2.3)
REQ-UC4-AC-4	Only the file owner should be able to authorize another user for reading or modifying his data files.	<ul style="list-style-type: none"> - Confidentiality - Integrity - Upload/download - Fine-grained retrieval - Write operations - Access by data owners - Selective sharing with other 	High		WP4 (T4.1)

		users/owners			
REQ-UC4-AC-5	Only the administrator of the system (the ESCUDO-CLOUD middleware) should be able to enable the access of a locked user.	<ul style="list-style-type: none"> - Confidentiality - Availability - Upload/download - Write operations - Access by data owners 	High	REQ-UC4-AC-6	WP4 (T4.1)
REQ-UC4-AC-6	Each user should have a limit of attempts to access the system.	<ul style="list-style-type: none"> - Confidentiality - Availability - Upload/download - Fine-grained retrieval - Write operations - Access by data owners 	High	REQ-UC4-AC-5	WP4 (T4.1)
REQ-UC4-SS-1	The cloud (or storage service) should have capacity enough to store the data files of the users.	<ul style="list-style-type: none"> - Availability - Upload/download - Write operations - Access by data owners - Single cloud provider - Multi clouds and federated clouds 	High	REQ-UC4-SS-3	WP4 (T4.1)
REQ-UC4-SS-2	The storage services should be accessible for the end users through ESCUDO-CLOUD middleware.	<ul style="list-style-type: none"> - Availability - Upload/download - Fine-grained retrieval - Write operations - Access by data owners - Multi clouds and federated clouds 	High		WP4 (T4.3)
REQ-UC4-SS-3	By using an elastic cloud should be possible to adapt	<ul style="list-style-type: none"> - Availability - Upload/download - Write operations 	High	REQ-UC4-SS-1	WP4 (T4.3)

	the capacity of the cloud to the user requirements.	<ul style="list-style-type: none"> - Access by data owners - Multi clouds and federated clouds 			
REQ-UC4-SS-4	The storage service should comply with “EU Directive 95/46/EC – The Data Protection Directive”.	<ul style="list-style-type: none"> - Confidentiality - Fine-grained retrieval Sharing - Single cloud provider - Multi clouds and federated clouds 	High	REQ-UC4-SS-5	WP2 (T2.1) WP4 (T4.1)
REQ-UC4-SS-5	The CSP should implement appropriate technical and organizational measures to protect personal data against accidental or unlawful destruction or accidental loss, alteration, unauthorized disclosure or access.	<ul style="list-style-type: none"> - Integrity - Upload/download - Write operations - Access by data owners - Single cloud provider - Multi clouds and federated clouds 	High	REQ-UC4-SS-4 REQ-UC4-DE-1 REQ-UC4-DE-2 REQ-UC4-DE-3 REQ-UC4-DE-4	WP2 (T2.1)
REQ-UC4-DE-1	The files should storage encrypted.	<ul style="list-style-type: none"> - Confidentiality - Integrity - Fine-grained retrieval - Access by data owners - Single cloud provider - Multi clouds and federated clouds 	High	REQ-UC4-DE-2 REQ-UC4-DE-3 REQ-UC4-SS-5	WP2 (T2.1)
REQ-UC4-DE-2	The data files should remain encrypted on the CSP until the user accesses them through a web	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Upload/download - Write operations - Access by data 	High	REQ-UC4-DE-1 REQ-UC4-SS-5	WP2 (T2.1)

	browser.	owners - Single cloud provider - Multi clouds and federated clouds			
REQ-UC4-DE-3	The data files should remain encrypted on the CSP until the user synchronizes the server with the agent installed on his device.	- Confidentiality - Integrity - Availability - Upload/download - Write operations - Access by data owners - Single cloud provider - Multi clouds and federated clouds	High	REQ-UC4-DE-1 REQ-UC4-SS-5	WP2 (T2.1)
REQ-UC4-DE-4	An end user should be able to securely download a data file from the CSP through the web browser.	- Confidentiality - Integrity - Availability - Upload/download - Write operations - Access by data owners - Single cloud provider - Multi clouds and federated clouds	High	REQ-UC4-SS-5	WP2 (T2.3) WP4 (T4.1)

6.4.7 Other Non-functional Requirements

6.4.7.1 Performance Requirements

The system should be responsive with minimal latency in:

- Authenticating the user.
- Loading list of accessible files and folders.
- Decrypting and downloading the selected file(s) to the users' device.

6.4.7.2 Compliance Requirements

- Encryption of files on cloud provider's storage should be suitably strong (e.g. AES-256). Equally, protection of these keys by ESCUDO-CLOUD should use strong encryption (e.g. RSA 2048-bit).

- An efficient protocol for the management of user keys and file/disk key should be in place so that the key management for large user and data sets with complex interacts does not become untenable.
- On termination of the transaction by the user, session timeout or loss of connection, any session information should be discarded by ESCUDO-CLOUD and the released file/disk keys should be destroyed. Unsaved changes by authorised users is lost in this instance too (ESCUDO-CLOUD cannot assume that the user intended to save changes).
- The user should have the possibility to ask for secure erase of his data (wiping procedure).

6.4.7.3 Software Quality Attributes

The following are common non-functional requirement for web pages that export services:

- Accessible – Accessible from different devices.
- Availability – 24 x 7 x 365. Service level agreements.
- Secured and Restorable – System and data backups. Business Continuity / Disaster Recovery.
- Certified – Web standard certifications, security and compliance certificates.
- Verifiable – System and process controls. How to know if the system is working.
- Documented – Level of required documentation.
- Open – APIs, system integration and interfaces.
- Quality – Identification and rectification of fault.
- Reliable – Consistent and dependable quality of service.
- Secure – Online assets need to be protected.
- Useable – Easy to use by target users. Both humans and web crawlers.

6.4.7.4 Business Rules

- Each user will belong to a role (depending on his profile), and each role will have different privileges.
- The agent will be installed in corporative computers.

6.4.7.5 Other Requirements

Any installation will be required to access to the stored files in the cloud through a web browser.

7 Analysis of ESCUDO-CLOUD Use Case Relationships and Requirements

7.1 Introduction

This chapter provides an analysis of the relationships between the use cases and their classification according to the ESCUDO-CLOUD dimensions. It forms our initial approach to the integration and synthesis of the requirements. The functional and non-functional requirements generated in this deliverable will be iteratively refined during the next phase of work and will be finalised by the end of month 12 and published in the form of project deliverable D1.2 as the final version of the security requirements of the ESCUDO-CLOUD use cases.

7.2 Use Case Relationships

As mentioned in the previous chapters, ESCUDO-CLOUD considers four different dimensions that help provide data owners with the combination of security and flexibility when trying to outsource their data resources to the cloud. All of the four use cases described in detail earlier, offer the data owners the capability of data protection on different storage mediums on cloud service providers. The ESCUDO-CLOUD dimensions address most of the areas of concern that have been identified while offering data owners user-centric outsourcing of their data to remote cloud storage services. Out of these areas, the security and privacy of the outsourced data is paramount and is addressed by all of the use cases as part of their “security properties”. The uses cases enforce this by using different encryption techniques to encrypt data file, block storage, object storage, relational databases and HDFS clusters.

Figure 37 shows an overview of the ESCUDO-CLOUD eco-system and the place and role of each use case in that eco-system. Together these four use cases cover the protection of data in a vast spectrum of data storage services being offered by most cloud service providers, ranging from file storage (WT), block storage (BT), object storage (BT/IBM), Big Data cluster storage (BT) and relational database management services (SAP). In addition to targeting different storage mediums, each use case also focuses on different types of protection, e.g., the IBM use case focuses on the server-side object storage encryption, the SAP use case focuses on searchable encryption of data in a relational database, BT use case focuses on encryption of block, object and cluster/distributed data-at-rest, and WT use case focusing on federated and secure access to encrypted files stored on different cloud service providers.

Figure 37 also shows the focus of each use case with respect to cloud elasticity, with IBM and SAP use cases focusing on single cloud service providers and BT and WT catering for multiple or federated cloud service providers. Moreover, some use cases also offer the data owners the capability to create policies and rules to allow other users to access their secure data from the cloud storage service. However, the common thread in all the use cases is that they provide the data owners with a managed service to protect their outsourced data, which can be stored on different types of cloud storage services, and use data encryption to enforce secure data access.

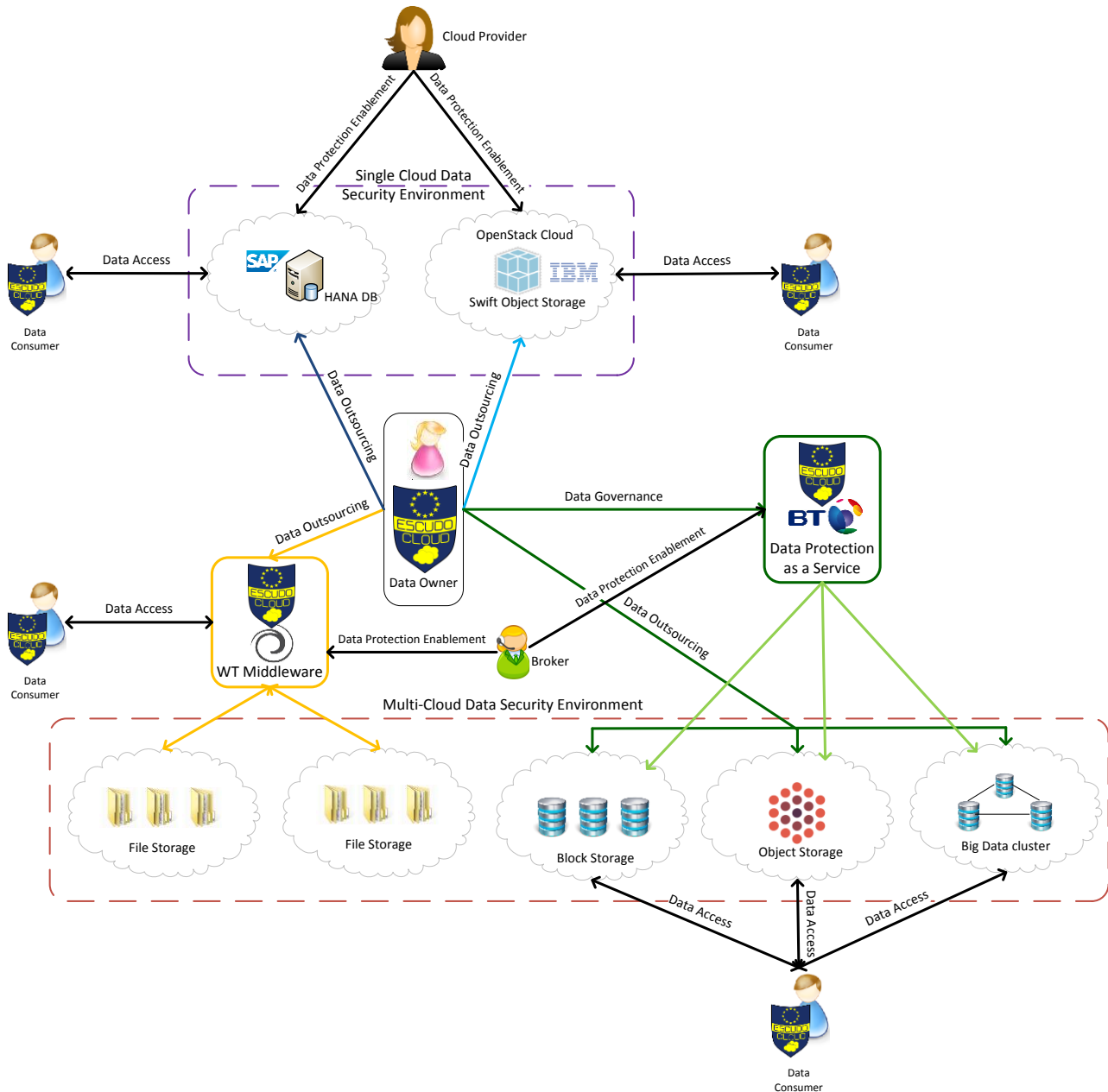


Figure 37. Overview of the Use Cases in the ESCUDO-CLOUD eco-system

In relation to targeting similar storage mediums, the IBM and BT use cases are similar in that each provides means of securing data on object storage services, however, the IBM use case focuses on

the server-side object encryption on OpenStack Swift in particular, whereas the BT use case focuses on using an encryption gateway located between the data owners and the cloud service provider to encrypt the objects on multiple object storage services in a multi-cloud scenario. Therefore, the IBM use case is focusing on provisioning the data security innovations on a single private cloud platform (OpenStack in this case), whereas the BT use case provisions the security as a service on multiple cloud platforms.

Similarly, both the BT and WT use cases offer data security solutions spanning multiple cloud platforms, however, the BT use case constructs this solution in form of a service offered through a service store on the the storage mediums of block, object and HDFS cluster storage, whereas the WT use case offers this solution as a middleware or broker service and focuses on file encryption in a multi-cloud scenario.

Another commonality that exists between some use cases exists with respect to cloud elasticity, with IBM and SAP use cases focusing on single cloud service providers and BT and WT catering for multiple or federated cloud service providers. However, all use cases offer data owners the capability to create rules and policies and allow other users access their secure data hosted the cloud storage service upon successful fulfillment of these rules and policies.

So the common feature in all the use cases is that they provide the data owners with a managed service to protect their outsourced data in the cloud environment and use data encryption to enforce secure data access. The differences are in the form of different types of cloud storage services, the level of multiplicity of cloud platforms, and the integration of key management and access management features with the security solutions.

7.3 Classification of Requirements

In this section we provide a view of the security requirements that have been obtained from the ESCUDO-CLOUD use cases and classify them according to each dimension.

7.3.1 Security Properties

In ESCUDO-CLOUD, security properties are related to the issues concerning confidentiality, integrity, and availability of data and resources pertaining to a data owner. Here we classify all the requirements from ESCUDO-CLOUD use cases that have to address this dimension.

7.3.1.1 USE CASE 1: OpenStack Framework

The requirements for addressing the security property issues in Use Case 1 are:-

- REQ-UC1-IKM-1: The system supports CRUD operations for cryptographic keys and related cryptographic material which is used in the cloud infrastructure.
- REQ-UC1-IKM-2: Deployment and management of infrastructure keys is driven by policies and automated, to the extent possible

- REQ-UC1-IKM-3: The cloud infrastructure-key management system supports the relevant open standards that are used by the industry (OASIS KMIP for REST APIs and possibly OASIS PKCS #11 for interfaces to secure hardware modules).
- REQ-UC1-IKM-4: The cloud infrastructure-key management system supports the secure deletion of cryptographic material.
- REQ-UC1-TKM-1: The system supports CRUD operations for tenant keys.
- REQ-UC1-TKM-2: Deployment and management of tenant keys is driven by policies and automated.
- REQ-UC1-TKM-3: The cloud tenant-key management system supports the relevant open standards that are used by the industry.
- REQ-UC1-TKM4: The cloud tenant-key management system supports the secure deletion of cryptographic material
- REQ-UC1-SKM-1: Key-management systems can be operated in a redundant and fault-tolerant way and do not introduce any single point of failure.
- REQ-UC1-SKM-2: Key-management systems do not limit the scalability of the cloud platform. They must either be offered with large enough throughput or they must be scalable on their own.
- REQ-UC1-SKM-3: Key-management systems can cope with the weak forms of consistency found in cloud platforms such as OpenStack. In particular, the key-management systems provide support for eventual consistency of the underlying operations in the cloud platform

7.3.1.2 USE CASE 2: Secure Enterprise Data Management in the Cloud

The requirements for addressing the security property issues in Use Case 2 are:-

- REQ-UC2-AC1: Access control decisions should be based on the subject of a client, i.e. the subject granularity is a client.
- REQ-UC2-AC2: It should be possible to group several users into a group and grant or revoke access for the entire group.
- REQ-UC2-AC3: Access control decisions should be based on the object of a database cell as identified by a column (in a table) and a row owner.
- REQ-UC2-AC4: The access control model should be an access control matrix.
- REQ-UC2-AC5: Access control rights should be grantable and revocable by the database administrator with support of the data owning clients.
- REQ-UC2-AC6: Access control should be enforced by the client, i.e. cryptographically enforced.
- REQ-UC2-KM1: Each client should have its own key generated and kept confidential at its site
- REQ-UC2-KM2: There should be keys for access by groups.
- REQ-UC2-KM3: Client keys should be stored securely, e.g. in a secure key store (PKCS#12) protected by a password.
- REQ-UC2-KM4: Group key may be derived using a public key hierarchy stored at the cloud service provider.

- REQ-UC2-EQ1: The database should support order-preserving encryption for range and rank queries, deterministic encryption for equality selection/joins, and grouping, probabilistic encryption for retrieval and count operations and additively homomorphic encryption for summation.
- REQ-UC2-EQ2: The encryption should be adjustable to the database operations performed.
- REQ-UC2-EQ3: The database driver should at least support three different query evaluation techniques: query rewriting, proxy re-encryption and post-processing.
- REQ-UC2-EQ4: All database operations should be supported across different client keys, i.e. spanning multiple access groups.

7.3.1.3 USE CASE 3: Federated Secure Cloud Storage

The requirements for addressing the security property issues in Use Case 3 are:-

- REQ-UC3-KM-1: Each tenant should be provisioned with an instance of a key management service from the cloud service store.
- REQ-UC3-KM-2: The tenants should be able to generate, modify and remove keys from their key management service instance.
- REQ-UC3-KM-3: The key management service should be able to offer different key types and generation algorithms to each tenant, e.g., AES128, AES256, 3DES etc.
- REQ-UC3-KM-4: Only the tenants should be able to create and manage the keys.
- REQ-UC3-KM-5: The cloud service providers should have no access or visibility of the tenants' keys.
- REQ-UC3-KM-6: The tenants should be able to cache their keys on trusted virtual machines or gateways in order to outsource or improve performance of the encryption and decryption process.
- REQ-UC3-AC-1: The tenants should be able to create, delete and modify access control policies from their instance of the access control service.
- REQ-UC3-AC-2: The tenants should be able to create, delete and modify access control policies from their instance of the access control service.
- REQ-UC3-AC-3: The access control service should be able to offer use of different system and data attributes for the construction of a security rule, e.g., filesystem, user, application, and time attributes.
- REQ-UC3-AC-4: Only the tenants should be able to create and manage their access control policies.
- REQ-UC3-AC-5: The cloud service providers should have no access or visibility of the tenants' keys.
- REQ-UC3-AC-6: All data protection operations should be governed by access control policies by either approving or denying access to the required keys.
- REQ-UC3-AC-7: The access control service of tenants should be tightly coupled with their key management service, such that no key can be utilised without an approving access control policy.

- REQ-UC3-DE-1: The core encryption process should only be controlled and managed by the tenant.
- REQ-UC3-DE-2: The tenant should be able to deploy and manage the core encryption process on trusted virtual machines or gateways as an agent or plug-in.
- REQ-UC3-DE-3: The core encryption process should be FIPS 140 compliant.
- REQ-UC3-DE-4: The encryption agent or plug-in should be able to access the tenant's key management service and access control service.
- REQ-UC3-DE-5: The keys should only be released to the encryption agent or plug-in upon approval of an access control policy.

7.3.1.4 USE CASE 4: Elastic Cloud Service Provider

The requirements for addressing the security property issues in Use Case 4 are:-

- REQ-UC4-AC-1: Each user should have a unique username and password to enter in the system through a web portal.
- REQ-UC4-AC-2: By using the agent, it should be possible to remember the credentials.
- REQ-UC4-AC-3: Each pair username/password should be associated with a specific role.
- REQ-UC4-AC-4: Only the file owner should be able to authorize another user for reading or modifying his data files.
- REQ-UC4-AC-5: Only the administrator of the system (the ESCUDO-CLOUD middleware) should be able to enable the access of a locked user.
- REQ-UC4-AC-6: Each user should have a limit of attempts to access the system.
- REQ-UC4-SS-1: The cloud (or storage service) should have capacity enough to store the data files of the users.
- REQ-UC4-SS-2: The storage services should be accessible for the end users through ESCUDO-CLOUD middleware.
- REQ-UC4-SS-3: By using an elastic cloud should be possible to adapt the capacity of the cloud to the user requirements.
- REQ-UC4-SS-4: The storage service should comply with "EU Directive 95/46/EC".
- REQ-UC4-SS-5: The CSP should implement appropriate technical and organizational measures to protect personal data against accidental or unlawful destruction or accidental loss, alteration, unauthorized disclosure or access, in particular where the processing involves the transmission of data over a network, and against all other unlawful forms of processing.
- REQ-UC4-DE-1: The files should be stored encrypted.
- REQ-UC4-DE-2: The data files should remain encrypted on the CSP until the user accesses them through a web browser.
- REQ-UC4-DE-3: The data files should remain encrypted on the CSP until the user synchronizes the server with the agent installed on his device.
- REQ-UC4-DE-4: An end user should be able to securely download a data file from the CSP through the web browser.

7.3.1.5 Common High-level Requirements



In this section we provide a summarised list of requirements that are similar across the four use cases with respect to the “security properties” dimension.

1. Key management service is required for the provisioning, management and deletion of encryption keys for all the use cases.
2. The life-cycle of the encryption keys should be governed by a policy-based access control service.
3. The data owner should have the ultimate control and visibility of the encryption keys and the data encryption process.
4. The key management and data encryption processes should conform to international interoperability and compliance standards.

7.3.2 Access Requirements

In ESCUDO-CLOUD, access requirements are related to the issues concerning accessing the data in bulk using the primitive upload/download I/O operations, or fine grained access to data based on selective queries, or access to data with write privileges to modify the data. Here we classify all the requirements from ESCUDO-CLOUD use cases that have to address this dimension.

7.3.2.1 USE CASE 1: OpenStack Framework

There are no requirements for addressing the access requirement issues in Use Case 1, as the Swift object storage service will be local to the OpenStack based cloud platform deployment.

7.3.2.2 USE CASE 2: Secure Enterprise Data Management in the Cloud

The requirements for addressing the access requirement issues in Use Case 2 are:-

- REQ-UC2-AC3: Access control decisions should be based on the object of a database cell as identified by a column (in a table) and a row owner.
- REQ-UC2-AC4: The access control model should be an access control matrix.
- REQ-UC2-EQ1: The database should support order-preserving encryption for range and rank queries, deterministic encryption for equality selection/joins, and grouping, probabilistic encryption for retrieval and count operations and additively homomorphic encryption for summation.
- REQ-UC2-EQ2: The encryption should be adjustable to the database operations performed.
- REQ-UC2-EQ3: The database driver should at least support three different query evaluation techniques: query rewriting, proxy re-encryption and post-processing.
- REQ-UC2-EQ4: All database operations should be supported across different client keys, i.e. spanning multiple access groups.

7.3.2.3 USE CASE 3: Federated Secure Cloud Storage

The requirements for addressing the access requirement issues in Use Case 3 are:-

- REQ-UC3-AC-7: The access control service of tenants should be tightly coupled with their key management service, such that no key can be utilised without an approving access control policy.

7.3.2.4 USE CASE 4: Elastic Cloud Service Provider

The requirements for addressing the access requirement issues in Use Case 4 are:-

- REQ-UC4-AC-1: Each user should have a unique username and password to enter in the system through a web portal.
- REQ-UC4-AC-2: By using the agent, it should be possible to remember the credentials.
- REQ-UC4-AC-3: Each pair username/password should be associated with a specific role.
- REQ-UC4-AC-4: Only the file owner should be able to authorize another user for reading or modifying his data files.
- REQ-UC4-AC-5: Only the administrator of the system (the ESCUDO middleware) should be able to enable the access of a locked user.
- REQ-UC4-AC-6: Each user should have a limit of attempts to access the system.
- REQ-UC4-SS-1: The cloud (or storage service) should have capacity enough to store the data files of the users.
- REQ-UC4-SS-2: The storage services should be accessible for the end users through ESCUDO-CLOUD middleware.
- REQ-UC4-SS-3: By using an elastic cloud should be possible to adapt the capacity of the cloud to the user requirements.
- REQ-UC4-SS-4: The storage service should comply with “EU Directive 95/46/EC”.
- REQ-UC4-SS-5: The CSP should implement appropriate technical and organizational measures to protect personal data against accidental or unlawful destruction or accidental loss, alteration, unauthorized disclosure or access, in particular where the processing involves the transmission of data over a network, and against all other unlawful forms of processing.
- REQ-UC4-DE-1: The files should be stored encrypted.
- REQ-UC4-DE-2: The data files should remain encrypted on the CSP until the user accesses them through a web browser.
- REQ-UC4-DE-3: The data files should remain encrypted on the CSP until the user synchronizes the server with the agent installed on his device.
- REQ-UC4-DE-4: An end user should be able to securely download a data file from the CSP through the web browser.

7.3.2.5 Common High-level Requirements

In this section we provide a summarised list of requirements that are similar across the four use cases with respect to the “access requirements” dimension of ESCUDO-CLOUD.

1. The data owners should be able to populate and retrieve keys to and from their key management solutions.

2. The data owners should have easy-to-use access interfaces to their outsourced data resources to perform actions equivalent to data insertion, retrieval and removal etc.
3. The data encryption and decryption processes should be as transparent to the user as possible.

7.3.3 Sharing Requirements

In ESCUDO-CLOUD, sharing requirements are related to the issues concerning keeping access to the data restricted to the data owner and/or sharing the data with users and entities other than the data owner in a selective and secure manner. Here we classify all the requirements from ESCUDO-CLOUD use cases that have to address this dimension.

7.3.3.1 USE CASE 1: OpenStack Framework

The requirements for addressing the sharing requirement issues in Use Case 1 are:-

- REQ-UC1-IKM-1: The system supports CRUD operations for cryptographic keys and related cryptographic material which is used in the cloud infrastructure.
- REQ-UC1-TKM-2: Deployment and management of tenant keys is driven by policies and automated.
- REQ-UC1-TKM-3: The cloud tenant-key management system supports the relevant open standards that are used by the industry.
- REQ-UC1-TKM4: The cloud tenant-key management system supports the secure deletion of cryptographic material.

7.3.3.2 USE CASE 2: Secure Enterprise Data Management in the Cloud

The requirements for addressing the sharing requirement issues in Use Case 2 are:-

- REQ-UC2-AC2: It should be possible to group several users into a group and grant or revoke access for the entire group.
- REQ-UC2-AC4: The access control model should be an access control matrix.
- REQ-UC2-AC5: Access control rights should be grantable and revocable by the database administrator with support of the data owning clients.
- REQ-UC2-KM2: There should be keys for access by groups.
- REQ-UC2-KM4: Group key may be derived using a public key hierarchy stored at the cloud service provider.

7.3.3.3 USE CASE 3: Federated secure cloud storage

The requirements for addressing the sharing requirement issues in Use Case 3 are:-

- REQ-UC3-KM-6: The tenants should be able to cache their keys on trusted virtual machines or gateways in order to outsource or improve performance of the encryption and decryption process.
- REQ-UC3-DE-2: The tenant should be able to deploy and manage the core encryption process on trusted virtual machines or gateways as an agent or plug-in.

- REQ-UC3-DE-4: The encryption agent or plug-in should be able to access the tenant's key management service and access control service.
- REQ-UC3-DE-5: The keys should only be released to the encryption agent or plug-in upon approval of an access control policy.

7.3.3.4 USE CASE 4: Elastic Cloud Service Provider

The requirements for addressing the sharing requirement issues in Use Case 4 are:-

- REQ-UC4-AC-1: Each user should have a unique username and password to enter in the system through a web portal.
- REQ-UC4-AC-2: By using the agent, it should be possible to remember the credentials.
- REQ-UC4-AC-3: Each pair username/password should be associated with a specific role.
- REQ-UC4-AC-4: Only the file owner should be able to authorize another user for reading or modifying his data files.
- REQ-UC4-AC-5: Only the administrator of the system (the ESCUDO middleware) should be able to enable the access of a locked user.
- REQ-UC4-AC-6: Each user should have a limit of attempts to access the system.
- REQ-UC4-SS-1: The cloud (or storage service) should have capacity enough to store the data files of the users.
- REQ-UC4-SS-2: The storage services should be accessible for the end users through ESCUDO-CLOUD middleware.
- REQ-UC4-SS-3: By using an elastic cloud should be possible to adapt the capacity of the cloud to the user requirements.
- REQ-UC4-SS-4: The storage service should comply with "EU Directive 95/46/EC".
- REQ-UC4-SS-5: The CSP should implement appropriate technical and organizational measures to protect personal data against accidental or unlawful destruction or accidental loss, alteration, unauthorized disclosure or access, in particular where the processing involves the transmission of data over a network, and against all other unlawful forms of processing.
- REQ-UC4-DE-1: The files should be stored encrypted.
- REQ-UC4-DE-2: The data files should remain encrypted on the CSP until the user accesses them through a web browser.
- REQ-UC4-DE-3: The data files should remain encrypted on the CSP until the user synchronizes the server with the agent installed on his device.
- REQ-UC4-DE-4: An end user should be able to securely download a data file from the CSP through the web browser.

7.3.3.5 Common High-level Requirements

In this section we provide a summarised list of requirements that are similar across the four use cases with respect to the "sharing requirements" dimension of ESCUDO-CLOUD.

1. The key management solutions should support the concept of group accounts where the members of the group have the feature of sharing or delegating encryption keys to other members.
2. The policy-based access control solutions should support the concept of group policies where the same policy can be applied to the members of a group.
3. The data owners should have the ability to share or delegate the encryption keys or the references there-of to the users they trust.

7.3.4 Cloud architectures

In ESCUDO-CLOUD, cloud architectures dimension is related to the issues concerning outsourcing of the data to a single cloud service provider, or multiple or federated cloud service providers. Here we classify all the requirements from ESCUDO-CLOUD use cases that have to address this dimension.

7.3.4.1 USE CASE 1: OpenStack Framework

There are no requirements for addressing the cloud architecture issues in Use Case 1, as it assumes all operations to be conducted in a single cloud environment.

7.3.4.2 USE CASE 2: Secure Enterprise Data Management in the Cloud

There are no requirements for addressing the cloud architecture issues in Use Case 2, as it assumes all operations to be conducted in a single cloud environment.

7.3.4.3 USE CASE 3: Federated Secure Cloud Storage

The requirements for addressing the cloud architecture issues in Use Case 3 are:-

- REQ-UC3-KM-1: Each tenant should be provisioned with an instance of a key management service from the cloud service store.
- REQ-UC3-KM-2: The tenants should be able to generate, modify and remove keys from their key management service instances.
- REQ-UC3-KM-5: The cloud service providers should have no access or visibility of the tenants' keys.
- REQ-UC3-KM-6: The tenants should be able to cache their keys on trusted virtual machines or gateways in order to outsource or improve performance of the encryption and decryption process.
- REQ-UC3-SO-2: The service store should provide the tenants with access to the storage services of multiple cloud service providers.
- REQ-UC3-SO-3: The service store should be able to offer block storage service to the tenants on multiple cloud service providers.
- REQ-UC3-SO-4: The service store should be able to offer object storage service to the tenants on multiple cloud service providers.

- REQ-UC3-SO-5: The service store should be able to offer Big Data storage service (HDFS) to the tenants on multiple cloud service providers.
- REQ-UC3-SO-6: The tenants should be able to enable or disable the use of data protection service on the storage service of their choice.
- REQ-UC3-SO-7: The service store should be able to offer key management as a service to the tenants.
- REQ-UC3-SO-8: The service store should be able to offer access control as a service to the tenants.

7.3.4.4 USE CASE 4: Elastic Cloud Service Provider

The requirements for addressing the cloud architecture issues in Use Case 4 are:-

- REQ-UC4-SS-1: The cloud (or storage service) should have capacity enough to store the data files of the users.
- REQ-UC4-SS-2: The storage services should be accessible for the end users through ESCUDO-CLOUD middleware.
- REQ-UC4-SS-3: By using an elastic cloud should be possible to adapt the capacity of the cloud to the user requirements.
- REQ-UC4-SS-4: The storage service should comply with “EU Directive 95/46/EC”.
- REQ-UC4-SS-5: The CSP should implement appropriate technical and organizational measures to protect personal data against accidental or unlawful destruction or accidental loss, alteration, unauthorized disclosure or access, in particular where the processing involves the transmission of data over a network, and against all other unlawful forms of processing.
- REQ-UC4-DE-1: The files should be stored encrypted.
- REQ-UC4-DE-2: The data files should remain encrypted on the CSP until the user accesses them through a web browser.
- REQ-UC4-DE-3: The data files should remain encrypted on the CSP until the user synchronizes the server with the agent installed on his device.
- REQ-UC4-DE-4: An end user should be able to securely download a data file from the CSP through the web browser.

7.3.4.5 Common High-level Requirements

In this section we provide a summarised list of requirements that are similar across the four use cases with respect to the “sharing requirements” dimension of ESCUDO-CLOUD.

1. The key management solutions should support and be able to work in a multi-cloud environment.
2. The policy-based access control solutions should support and be able to work in a multi-cloud environment.
3. The conceptual grouping of users and policies discussed earlier should remain applicable in a multi-cloud environment.

4. The transparent encryption, and data and key sharing requirements described earlier should remain in place in the multi-cloud environment.
5. The data owners should have data and key management interfaces that are consolidated or integrated for operation in a multi-cloud environment.

8 Conclusions and Next Steps

In this document we have provided the initial set of requirements for the four high-level business use cases considered by the ESCUDO-CLOUD project. We have elaborated each high-level use case in terms of representative scenarios that cover the key elements of the problems being addressed in the project. To gather the requirements from these scenarios, we decomposed each high-level use case into technical use cases that describe the actors involved in the system and the operations they need to perform in order to complete an action. The project followed an agreed methodology based on the standard practice of defining the purpose of each technical use case, assigning a priority to it in relation to its objectives and detailing lists of pre-conditions, sequence of events and post-conditions required for realising the technical use case. This helped us in building an initial solution architecture for each use case. To elicit the functional and non-functional requirements required for implementing this architecture, we performed a detailed analysis of the system features of each use case and catalogued the functional requirements so that they are easily accessible and usable for the work packages and tasks of ESCUDO-CLOUD that need them.

The initial functional and non-functional requirements obtained as a result of the exercise carried out in this deliverable will be used to drive the research and development work in the technical core work packages WP2, WP3 and WP4. These requirements will ensure that the work carried out in technical tasks of the ESCUDO-CLOUD project corresponds to actual needs of the use cases and enables direct exploitation of that work by the industrial partners.

The technical results taken from WPs 2, 3 and 4 will then be fed back into WP 1 to be validated through the development of prototype applications, demonstrating the applicability and advantages of the novel solutions developed in the project within each use case. In particular, for solutions which are aimed at Technology Readiness Level 7 (TRL7) whose preliminary versions will be produced in WPs 2, 3 and 4.

Lastly, the functional and non-functional requirements generated in this deliverable will be iteratively refined again after six months as work on the technical core WPs of the project will have been underway by then. Besides refining the specific technical requirements of each use case, the refinement to be documented in D1.2 will include eliciting higher level security requirements that stem from multiple lower level technical requirements in one or multiple use-cases of this report. These elicited requirements will inform research and innovation in the corresponding tasks of the ESCUDO-CLOUD core WPs. The elicited requirements in D1.2 will also have associated success and validation criteria that will in turn inform the future validation tasks of WP1. This will help in

steering or correcting the entire technological development process of the use cases according to the project goals and objectives. The refinements of requirements from the technical use cases will be finalised by the end of month 12 and published in the form of project deliverable D1.2 as the final version of the security requirements of the ESCUDO-CLOUD use cases.

9 Bibliography

- [1] Google Docs, <http://www.google.com/docs>
- [2] Dropbox, <https://www.dropbox.com/>
- [3] SpiderOak, <https://spideroak.com/>
- [4] Tresorit, <https://www.tresorit.com/>
- [5] Article 29 Data Protection Working Party, Opinion 05/2012 on Cloud Computing
- [6] ESCUDO-CLOUD: Description of the Action
- [7] IDC Report: <http://www.idc.com/getdoc.jsp?containerId=prUS24977214t>
- [8] A Guide to the Business Analysis Body of Knowledge. International Institute of Business Analysis. 2009. [ISBN 978-0-9811292-1-1](#)

Annex A: ESCUDO-CLOUD Integrated Requirements Catalogue

The consolidated and summarised security requirements of all the ESCUDO-CLOUD use cases are given in the following table.

Table 6. Integrated Requirements Catalogue of ESCUDO-CLOUD

Requirement Reference #	Requirement Description	ESCUDO-CLOUD dimension	Priority	Dependencies on other Requirement	Relevant Core Work Package
REQ-UC1-IKM-1	CRUD operations for infrastructure keys	- Confidentiality - Availability	Medium		WP2 (T2.1, T2.2)
REQ-UC1-IKM-2	Policy-driven and automated infrastructure-key management	- Confidentiality - Availability - Fine-grained retrieval	Medium	REQ-UC1-IKM-1	WP2 (T2.1, T2.2)
REQ-UC1-IKM-3	Support for standard APIs and protocols in infrastructure-key management	- Single cloud provider	Medium	REQ-UC1-IKM-1	WP2 (T2.2)
REQ-UC1-IKM4	Support for secure deletion of cryptographic material	- Confidentiality	Medium	REQ-UC1-IKM-2, REQ-UC1-IKM3	WP2 (T2.1)

REQ-UC1-TKM-1	CRUD operations for tenant keys	- Confidentiality - Availability	High		WP2 (T2.2)
REQ-UC1-TKM-2	Policy-driven and automated tenant-key management	- Confidentiality - Availability - Fine-grained retrieval	Medium	REQ-UC1-TKM-1	WP2 (T2.2)
REQ-UC1-TKM-3	Support for standard APIs and protocols in tenant-key management	- Single cloud provider	High	REQ-UC1-TKM-1	WP2 (T2.2)
REQ-UC1-TKM4	Support for secure deletion of cryptographic material	- Confidentiality	High	REQ-UC1-TKM-2, REQ-UC1-TKM-3	WP2 (T2.1)
REQ-UC1-SKM-1	Redundancy and fault-tolerance in key-management systems	- Availability	High	REQ-UC1-TKM-1, REQ-UC1-IKM1	WP2 (T2.2)
REQ-UC1-SKM-2	Scalable design of key-management system	- Confidentiality - Availability	Medium	REQ-UC1-TKM-1, REQ-UC1-IKM1	WP2 (T2.2)
REQ-UC1-SKM-3	Key-management solutions support weakly consistent operations in cloud platform	- Confidentiality - Availability - Selective sharing with other users/owners	Medium	REQ-UC1-TKM-1, REQ-UC1-IKM1, REQ-UC1-SKM-1	WP2 (T2.2) WP3 (T3.2)
REQ-UC2-AC1	Access control per client	- Confidentiality - Access by data owners	High	REQ-UC2-KM1	WP2

REQ-UC2-AC2	Access control per group of clients	<ul style="list-style-type: none"> - Confidentiality - Selective sharing with other users/owners 	High	REQ-UC2-KM2	WP3
REQ-UC2-AC3	Access control per database cell	<ul style="list-style-type: none"> - Confidentiality - Fine-grained retrieval 	High	REQ-UC2-EQ2	WP3
REQ-UC2-AC4	Access control matrix model	<ul style="list-style-type: none"> - Confidentiality - Access by data owners - Selective sharing with other users/owners - Fine-grained retrieval 	Medium		WP2/WP3
REQ-UC2-AC5	Access grant and revoke by administrator	<ul style="list-style-type: none"> - Confidentiality - Integrity - Selective sharing with other users/owners 	Low	REQ-UC2-EQ3	WP2/WP3
REQ-UC2-AC6	Access control enforced by client	<ul style="list-style-type: none"> - Confidentiality - Integrity 	High	REQ-UC2-KM3	WP2/3
REQ-UC2-KM1	One key per client	<ul style="list-style-type: none"> - Confidentiality - Integrity - Access by data owners 	High		WP2
REQ-UC2-KM2	Group key management	<ul style="list-style-type: none"> - Confidentiality - Integrity - Selective sharing with other users/owners 	High		WP2/WP3

REQ-UC2-KM3	Client key securely stored at client only	- Confidentiality - Integrity - Access by data owners	High		WP2
REQ-UC2-KM4	Group keys derivable	- Confidentiality - Integrity - Selective sharing with other users/owners	High	REQ-UC2-KM2	WP2/WP3
REQ-UC2-EQ1	Encryption schemes	- Confidentiality - Fine-grained retrieval - Write operations	High		WP3
REQ-UC2-EQ2	Adjustable onion encryption	- Confidentiality - Fine-grained retrieval - Write operations	High	REQ-UC2-EQ1	WP3
REQ-UC2-EQ3	Proxy re-encryption, Query rewriting, Post-processing	- Confidentiality - Fine-grained retrieval - Write operations	High		WP3
REQ-UC2-EQ4	Support for different keys	- Confidentiality - Fine-grained retrieval - Write operations	High	REQ-UC2-EQ3	WP3
REQ-UC2-AC1	Access control per client	- Confidentiality - Access by data owners	High	REQ-UC2-KM1	WP2
REQ-UC2-AC2	Access control per group of clients	- Confidentiality - Selective sharing with other users/owners	High	REQ-UC2-KM2	WP3

REQ-UC2-AC3	Access control per database cell	<ul style="list-style-type: none"> - Confidentiality - Fine-grained retrieval 	High	REQ-UC2-EQ2	WP3
REQ-UC2-AC4	Access control matrix model	<ul style="list-style-type: none"> - Confidentiality - Access by data owners - Selective sharing with other users/owners - Fine-grained retrieval 	Medium		WP2/WP3
REQ-UC2-AC5	Access grant and revoke by administrator	<ul style="list-style-type: none"> - Confidentiality - Integrity - Selective sharing with other users/owners 	Low	REQ-UC2-EQ3	WP2/WP3
REQ-UC2-AC6	Access control enforced by client	<ul style="list-style-type: none"> - Confidentiality - Integrity 	High	REQ-UC2-KM3	WP2/3
REQ-UC2-KM1	One key per client	<ul style="list-style-type: none"> - Confidentiality - Integrity - Access by data owners 	High		WP2
REQ-UC2-KM2	Group key management	<ul style="list-style-type: none"> - Confidentiality - Integrity - Selective sharing with other users/owners 	High		WP2/WP3
REQ-UC2-KM3	Client key securely stored at client only	<ul style="list-style-type: none"> - Confidentiality - Integrity - Access by data owners 	High		WP2

REQ-UC2-KM4	Group keys derivable	<ul style="list-style-type: none"> - Confidentiality - Integrity - Selective sharing with other users/owners 	High	REQ-UC2-KM2	WP2/WP3
REQ-UC2-EQ1	Encryption schemes	<ul style="list-style-type: none"> - Confidentiality - Fine-grained retrieval - Write operations 	High		WP3
REQ-UC2-EQ2	Adjustable onion encryption	<ul style="list-style-type: none"> - Confidentiality - Fine-grained retrieval - Write operations 	High	REQ-UC2-EQ1	WP3
REQ-UC2-EQ3	Proxy re-encryption, Query rewriting, Post-processing	<ul style="list-style-type: none"> - Confidentiality - Fine-grained retrieval - Write operations 	High		WP3
REQ-UC2-EQ4	Support for different keys	<ul style="list-style-type: none"> - Confidentiality - Fine-grained retrieval - Write operations 	High	REQ-UC2-EQ3	WP3
REQ-UC3-KM-1	Each tenant should be provisioned with an instance of a key management service from the cloud service store	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Access by data owners - Multi clouds 	High	REQ-UC3-AC-7 REQ-UC3-SO-1 REQ-UC3-SO-2	WP 2 (T2.2)
REQ-UC3-KM-2	The tenants should be able to generate, insert, retrieve and remove keys from their key management service	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Access by data owners - Upload/Download - Multi clouds 	Medium	REQ-UC3-KM-1	WP 2 (T2.2)

REQ-UC3-KM-3	The key management service should be able to offer different key types and generation algorithms to each tenant, e.g., AES128, AES256, 3DES etc.	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability 	Low	REQ-UC3-KM-1 REQ-UC3-DE-1 REQ-UC3-DE-3	WP 2 (T2.2)
REQ-UC3-KM-4	Only the tenants should be able to create and manage the keys	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Access by data owners 	High	REQ-UC3-KM-1	WP 2 (T2.3)
REQ-UC3-KM-5	The cloud service providers should have no access or visibility of the tenants' keys	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Access by data owners - Multi clouds 	High	REQ-UC3-KM-1	WP 2 (T2.3)
REQ-UC3-KM-6	The tenants should be able to cache their keys on trusted virtual machines or gateways in order to outsource or improve performance of the encryption and decryption process	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Selective sharing with other users/owners - Multi clouds 	Low	REQ-UC3-KM-1 REQ-UC3-DE-2	WP 3 (T3.1)
REQ-UC3-AC-1	Each tenant should be provisioned with an instance of an access control service from the cloud service store	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Access by data owner - Multi clouds 	High	REQ-UC3-SO-1 REQ-UC3-SO-6	WP 2 (T2.3)
REQ-UC3-AC-2	The tenants should be able to create, delete and modify access control policies from their instance of the access control service	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Access by data owner 	Medium	REQ-UC3-AC-1	WP 2 (T2.3)

REQ-UC3-AC-3	The access control service should be able to offer use of different system and data attributes for the construction of a security rule, e.g., filesystem, user, application, and time attributes.	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability 	Low	REQ-UC3-AC-1	WP 2 (T2.3)
REQ-UC3-AC-4	Only the tenants should be able to create and manage their access control policies	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Access by data owner 	High	REQ-UC3-AC-1	WP 2 (T2.3)
REQ-UC3-AC-5	The cloud service providers should have no access or visibility of the tenants' access control policies	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Access by data owner 	High	REQ-UC3-AC-1	WP 2 (T2.3)
REQ-UC3-AC-6	All data protection operations should be governed by access control policies by either approving or denying access to the required keys	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability 	High	REQ-UC3-AC-1 REQ-UC3-KS-1	WP 2 (T2.3)
REQ-UC3-AC-7	The access control service of tenants should be tightly coupled with their key management service, such that no key can be utilised without an approving access control policy	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Fine grained retrieval 	High	REQ-UC3-AC-1 REQ-UC3-KM-1	WP 2 (T2.3)
REQ-UC3-SO-1	Each tenant should be provisioned with a cloud service store account	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Access by data owner - Single cloud provider 	High		WP 4 (T4.3)

REQ-UC3-SO-2	The service store should provide the tenants with access to the storage services of multiple cloud service providers	- Multi clouds and federated clouds	Medium		WP 4 (T4.3)
REQ-UC3-SO-3	The service store should be able to offer block storage service to the tenants	- Multi clouds and federated clouds	High		WP 4 (T4.3)
REQ-UC3-SO-4	The service store should be able to offer object storage service to the tenants	- Multi clouds and federated clouds	High		WP 4 (T4.3)
REQ-UC3-SO-5	The service store should be able to offer Big Data storage service (HDFS) to the tenants	- Multi clouds and federated clouds	High		WP 4 (T4.3)
REQ-UC3-SO-6	The tenants should be able to enable or disable the use of data protection service on the storage service of their choice	- Multi clouds and federated clouds	Medium		WP 4 (T4.3)
REQ-UC3-SO-7	The service store should be able to offer key management as a service to the tenants	- Multi clouds and federated clouds	High		WP 4 (T4.3)
REQ-UC3-SO-8	The service store should be able to offer access control as a service to the tenants	- Multi clouds and federated clouds	High		WP 4 (T4.3)
REQ-UC3-DE-1	The core encryption process should only be controlled and managed by the tenant	- Confidentiality - Integrity - Availability - Access by data owner	High		WP 2 (T2.1) WP 2 (T2.3)

REQ-UC3-DE-2	The tenant should be able to deploy and manage the core encryption process on trusted virtual machines or gateways as an agent or plug-in	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Selective sharing with other users/owners 	Medium		WP 2 (T2.3) WP 3 (T3.1)
REQ-UC3-DE-3	The core encryption process should be FIPS 140 compliant	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability 	Medium		WP 2 (T2.1)
REQ-UC3-DE-4	The encryption agent or plug-in should be able to access the tenant's key management service and access control service	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Selective sharing with other users/owners 	Medium		WP 2 (T2.3) WP 3 (T3.1)
REQ-UC3-DE-5	The keys should only be released to the encryption agent or plug-in upon approval of an access control policy	<ul style="list-style-type: none"> - Confidentiality - Integrity - Availability - Selective sharing with other users/owners 	High		WP 2 (T2.3) WP 3 (T3.1)
REQ-UC4-AC-1	Each user should have a unique username and password to enter in the system through a web portal.	Confidentiality Upload/download Fine-grained retrieval Write operations Access by data owners	High		WP2 (T2.3)
REQ-UC4-AC-2	By using the agent, it should be possible to remember the credentials.	Confidentiality Upload/download Fine-grained retrieval Write operations Access by data owners	High		WP2 (T2.3)
REQ-UC4-AC-3	Each pair username/pass should be associated with a specific role.	Confidentiality Upload/download Fine-grained retrieval Write operations Access by data owners	High		WP2 (T2.3)

REQ-UC4-AC-4	Only the file owner should be able to authorize another user for reading or modifying his data files.	Confidentiality Integrity Upload/download Fine-grained retrieval Write operations Access by data owners Selective sharing with other users/owners	High		WP4 (T4.1)
REQ-UC4-AC-5	Only the administrator of the system (the ESCUDO-CLOUD middleware) should be able to enable the access of a locked user.	Confidentiality Availability Upload/download Write operations Access by data owners	High	REQ-UC4-AC-6	WP4 (T4.1)
REQ-UC4-AC-6	Each user should have a limit of attempts to access the system.	Confidentiality Availability Upload/download Fine-grained retrieval Write operations Access by data owners	High	REQ-UC4-AC-5	WP4 (T4.1)
REQ-UC4-SS-1	The cloud (or storage service) should have capacity enough to store the data files of the users.	Availability Upload/download Write operations Access by data owners Single cloud provider Multi clouds and federated clouds	High	REQ-UC4-SS-3	WP4 (T4.1)
REQ-UC4-SS-2	The storage services should be accessible for the end users through ESCUDO-CLOUD middleware.	Availability Upload/download Fine-grained retrieval Write operations Access by data owners Multi clouds and federated clouds	High		WP4 (T4.3)
REQ-UC4-SS-3	By using an elastic cloud should be possible to adapt the capacity of the cloud to the user requirements.	Availability Upload/download Write operations Access by data owners Multi clouds and federated clouds	High	REQ-UC4-SS-1	WP4 (T4.3)

REQ-UC4-SS-4	The storage service should comply with “EU Directive 95/46/EC – The Data Protection Directive”.	Confidentiality Fine-grained retrieval Sharing Single cloud provider Multi clouds and federated clouds	High	REQ-UC4-SS-5	WP2 (T2.1) WP4 (T4.1)
REQ-UC4-SS-5	The CSP should implement appropriate technical and organizational measures to protect personal data against accidental or unlawful destruction or accidental loss, alteration, unauthorized disclosure or access.	Integrity Upload/download Write operations Access by data owners Single cloud provider Multi clouds and federated clouds	High	REQ-UC4-SS-4 REQ-UC4-DE-1 REQ-UC4-DE-2 REQ-UC4-DE-3 REQ-UC4-DE-4	WP2 (T2.1)
REQ-UC4-DE-1	The files should storage encrypted.	Confidentiality Integrity Fine-grained retrieval Access by data owners Single cloud provider Multi clouds and federated clouds	High	REQ-UC4-DE-2 REQ-UC4-DE-3 REQ-UC4-SS-5	WP2 (T2.1)
REQ-UC4-DE-2	The data files should remain encrypted on the CSP until the user accesses them through a web browser.	Confidentiality Integrity Availability Upload/download Write operations Access by data owners Single cloud provider Multi clouds and federated clouds	High	REQ-UC4-DE-1 REQ-UC4-SS-5	WP2 (T2.1)
REQ-UC4-DE-3	The data files should remain encrypted on the CSP until the user synchronizes the server with the agent installed on his device.	Confidentiality Integrity Availability Upload/download Write operations Access by data owners Single cloud provider Multi clouds and	High	REQ-UC4-DE-1 REQ-UC4-SS-5	WP2 (T2.1)

		federated clouds			
REQ-UC4-DE-4	An end user should be able to securely download a data file from the CSP through the web browser.	Confidentiality Integrity Availability Upload/download Write operations Access by data owners Single cloud provider Multi clouds and federated clouds	High	REQ-UC4-SS-5	WP2 (T2.3) WP4 (T4.1)