HORIZON 2020

The EU Framework Programme for Research and Innovation

prisma cloud

Privacy and Security Maintaining Services in the Cloud
Contract Number: 644962
Call: H2020-ICT-2014-1

DOMAIN INDEPENDENT

GENERIC SECURITY MODELS

Deliverable D 2.2

Deliverable due date: 31.01.2016

Document Information	
Title	Domain independent generic security models
Creator	Thomas Länger, UNIL
Deliverable no.	2.2
Work Package No.	2
Nature	Report
Dissemination Level	PU
Date	31 Jan 2016 (t0+12)
Document description	-

Authors List

Organization	Name	E-mail	Phone Number
UNIL	Thomas Länger	thomas.laenger@unil.ch	
XT	Katrin Riemer	katrin.riemer@xitrust.com	
ATOS	Ángel Palomares Pérez	angel.palomares@atos.net	
IRT	Matteo Biancani	matteo.biancani@interoute.com	
UNI PASSAU	Henrich C. Pöhls	hp@sec.uni.passau.de	

Reviewers List

Organization	Name	E-mail
LISPA	Sara Faccinetti Marco Decandia Brocca	sara.facchinetti@cnt.lispa.it marcodec.lispa@gmail.com
UNI PASSAU	Henrich C. Pöhls[1]	hp@sec.uni.passau.de
KAU	Simone Fischer-Hübner (additional reviewer)	simone.fischer-huebner@kau.se

[11] Henrich C. Poehls is unfortunately both author and (internal) reviewer. UNI PASSAU was not planned as contributor for task T2.2 but later joined in, mainly for two cloud security patterns, and did also a lot of other work for the deliverable. Actually the project management should have been informed by the task leader (me, Thomas Länger, UNIL) that they appoint another reviewer—but this did not happen.

Version history

Version	Date	Reason/Change	Editor
V01	16-Jun-15	Initial version	Thomas Länger
V02	23-Jun-15	Modification to structure, started introduction	“
V03	10-Jul-15	Modification/structure, first distribution of tasks	“
V04	3-Sep-15	Routine update	“
V05	6-Oct-15	Routine update	“
V06	22-Oct-15	Added preliminary chapter 6	“
V07	27-Oct-15	Routine update	“
V08	05-Nov-15	New adapted TOC	“
V09	03-Dec-15	Now throwing all the old stuff out, fill new content at the beginning (introduction, methodology); try to manifest the new structure from V08… As well trying to implement the “plan” of Marco and Sara	“
V09	22-Nov-15	Still V09 but significantly enhanced (3. Introduction; 5. Taxonomies; 7. Crypto functions & patterns)	“
V10	24-Nov-15	Many changes, starting the patterns	“
V11	03-Dec-15	Added new IRT 3.1 and 5	“
V12	15-Dec-15	Entire patterns chapter has new structure; named all patterns; added new contents by IRT/Paolo (entire 2 and entire 5)	“
V13	10-Jan-16	Internal review version	“
V14 =v1.0	31-Jan-16	Delivered version	“

Table of Contents

1 Abstract

This document is D2.2 “Domain independent generic security models” of task T2.2 “Refine and analyse domain independent generic requirements and security goals”. D2.2 is one of four main deliverables from work package WP2 “Use cases and requirements”. In addition to the “Legal, social and HCI[2] requirements” of D2.1, and the detailed description of the use cases on which the new PRISMACLOUD cloud security functions will be demonstrated (D2.3 “Use case specification”), and the “Risk and threat analysis with security requirements” of D2.5, this document develops the **generic situations in cloud usage, where security and privacy problems occur—and where the PRISMACLOUD functions can be applied to mitigate those problems.** The situations are specifically regarded from a cloud customer or end user perspective.

We start with an assessment on how **security and privacy** is regarded in **current cloud services** and applications. To this goal, current cloud ontologies and reference architectures are being analysed, and the privacy policies, the **privacy guarantees, and other security options of the major cloud provider**s investigated in depth and compared in a synoptic table. This analysis is followed by an exploration of major **security benefits and security risks in cloud computing**. The analysis of the current situation frames the context for the presentation of eight cloud security patterns of situations, which occur over again in public cloud environments—situations where the end user security or privacy is challenged and often compromised. The **eight cloud security patterns** describe situations where the **application of PRISMACLOUD cryptographic primitives** can significantly improve the security of the end user, or protect his/her privacy better than current solutions.

The **cloud security patterns shall be re-used in the “Security and privacy by design” task** of WP7 “Composition of next-generation secure cloud services” to provide guidance for the scientists and engineers working on the development and implementation of the cryptographic primitives, as well as to communicate the potential and the capabilities of the PRISMACLOUD crypto primitives to end users.

[2] Human Computer Interaction

2 Introduction and scope

2.1 Introduction

Cloud computing is the major growth area in information and communication technologies today, and with its huge processing capabilities and data storage architectures, and with all the data which is amassed, and even created through its use, it is closely related to another major growth area in ICT, that of big data aggregation, processing and analysis. With an estimated size of about 150 billion US-Dollar, the cloud market is highly contested, and large corporations push with enormous effort into this market. Today's biggest players are in fact companies which have enormous financial power at their disposal and are proficiently experienced in the field of ICT. They generated their enormous wealth already in the (virtual) worlds of the Internet and are now hurrying to even increase revenue and power in the developing information age by investing huge effort in the field of cloud computing. The biggest cloud provider to-date, Amazon.com Inc., started as online book store in 1994 and generated (and is still generating) enormous wealth as E-commerce retailer. The second and third biggest cloud providers are Microsoft and Google, who made their fortunes in PC operating systems and office software, and in search engines and internet advertising business, respectively. Besides the mentioned three cloud providers, there are several other providers and players in the field with comparable size and market share.

So there is currently an enormous rush into cloud computing and many of the commercial stakeholders are hurrying to stake their claims early and quickly gain a substantial market share, and thus a competitive edge in this profitable and steeply growing sector. In the history of ICT innovation several comparable situations are known, when companies rushed into a newly developing market, while at the same time also shaping the market. In such a hurry, developments often do not respect the requirements and needs of the end users—but rather the needs of the companies, which want to grow quickly. The price in these situations is often paid by the end-users. Systems and services are made available on a large scale before the privacy and data security concerns of the customers are fully addressed and resolved. Security breaches with considerable negative impact are frequent, and it is not clear, if the situation is currently improving or worsening. The risk is often bestowed upon the cloud users, because current cloud systems seem to value functionality more than end-user security. This includes legal risks for end users, with the disadvantage of being the initially legally liable party.

Nevertheless, this situation is already changing, and will likely change more in the nearer future, because there are huge market sectors of security aware customers who are currently barred from moving into the cloud—be it because they are forced by regulation to guarantee a certain degree of security for the data they are operating with (e.g. in the health sector or in electronic government), or they are just companies, or even individuals, who highly value the confidentiality of their data. The big cloud providers, are not only after the financial revenue—they seek to extend their spheres of influence and political power towards the control of these sensitive sectors. Here they do not only extract valuable information from the metadata which inevitably occurs in transactions and processing in the cloud, but in many cases also want to have access to the sensitive data itself,

to add it to the huge data collections they are already having of individuals from their search engines (Google) and smartphone operating system (Google, Apple), their market web sites (Amazon), from their social networks and from other sources on the Internet.

The European Commission, in its endeavour to strengthen European competitiveness and in its struggle to maintain European sovereignty over the data which is being moved to the cloud, has developed a proprietary European Cloud Computing Strategy [1], and supports the development of secure cloud systems in their Horizon 2020 strategic programme, of which PRISMACLOUD is a part of. The Commission recognizes the enormous cost reduction potential for companies of all sizes, which can be realised by a move to the cloud. Foremost, it recognizes the strategic importance of a European share and participation in the development and commercialisation of cloud computing products and services, and what is more, the strategic importance of maintaining sovereignty by not losing "European data" to opaque conglomerates beyond European data protection legislation and control.

Whether European research and development will be able to economically contest with its U.S. American competitors however remains questionable: Already now, almost the entire cloud business has its headquarters in the United States of America, in the area of Seattle, Washington (Microsoft-Skype and Amazon) and in California in the San Francisco Bay area, in the Silicon Valley (Google-Android-YouTube, Apple-iCloud, Facebook, DropBox and thousands of other companies). It is also there, and in huge data centres all across the United States, where the clouds are physically hosted, and the data is stored and processed[3]. In the USA we can currently witness how the Information and Communication technology sector (ICT) converges with the huge business of aggregating, processing, and commercialising today's data flood. Huge economic powers have emerged, which are reaching for total information control. European industries compete in the shadow of the American market giants, like in many other major fields of ICT. Yet, the European Commission sees an opportunity to focus on original European strengths of data security and privacy protection for the benefit of the end-users and customers.

The PRISMACLOUD project wants to provide advanced tools for implementing privacy and security services in the cloud, where the end-users remain in full control of their assets, and where user privacy is respected, in the sense that the amount of information and data, which is not essential to the operation of an application or a service, which is leaking or newly created by the operation, is minimised. One of the project goals is to build the security and privacy into the systems from the beginning—by design. Several requirements for such systems have already been elicited in other deliverables of the project (D2.1 Legal, social and HCI requirements and D2.3 Use

[3] It is now, that cloud providers have started to host their data centers in multiple locations world-wide, including Asia, South America, and countries of the European Union (see e.g. Amazon: http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-regions-availability-zones.html). Nevertheless, the headquarters and main installations of these businesses is certainly in the U.S.A. and it is at least probable that data, in whichever form and state of aggregation, is consolidated with data residing in the U.S.A. With the coming adoption of the new European General Data Protection Regulation (GDPR) in early 2016, the situation may again be subject to change.

case specification). We will here again elicit requirements: we will, very specifically from an end-user perspective, try to identify generic patterns of "situations" with relevance for the information security and privacy of personal and critical data in cloud systems. We want to describe the security and privacy concerns of end-users in recurring standard situations, in order to provide guidance and requirements for the design and implementation of the security mechanism itself.

2.2 Research problems and contributions

We argue that end-user security and privacy is not sufficiently addressed in current cloud paradigms and implementations. Current cloud ontologies and reference architectures tend to regard technical cloud systems mainly from the functional perspective of a service provider or application provider, or from the point of view of a cloud system supplier—thus neglecting important requirements of the cloud customer or end user. End user security is often only added in a layered manner outside the central function of the cloud system, while the core remains unprotected. There exist standards for contractual liability clauses to cope with the risk of data breaches in insufficiently protected systems. We will assess the significance of security in **current cloud ontologies and reference architectures**. It is most astonishing, that current cloud ontologies seem to omit the long-known fact in IT-security, that security objectives are usually of a relative nature per stakeholder, and that there can be conflicting security objectives in one system (e.g. user privacy versus state surveillance etc.). We will search for evidence on how different, potentially conflicting security objectives are regarded in current ontologies and reference architectures.

In the same way, as ontologies and reference architectures, through their normative character, have an influence on the information systems which are being developed and deployed in the cloud, **taxonomies of cloud computing applications and services** exert an influence by shaping and confining categories of cloud systems. We want to verify, if our perception can be substantiated. Our perception is that common taxonomies, namely we argue for the ontologies and reference architectures, omit a proper security and privacy perspective. From an end user security and privacy standpoint, we assume that, for example, a classification according to who has sovereignty over the data can be relevant.

To complete the framing of the context, we will list the most relevant **security and privacy threats** which prevail for end users in the cloud. We will not carry out the survey from scratch—that work has been done already several times by information security agencies, standardization organisations and in research projects. There are comprehensive threat and risk catalogues available—but we will focus mainly on a risk catalogue provided by the European Union Agency for Network and Information Security (ENISA), which is supporting the European Commission in the implementation of its European Cloud Computing Strategy.

The PRISMACLOUD project proposal identifies several challenges in current cloud systems and service offerings, which effectively stand against the deployment of sensitive data and applications to the cloud. These challenges include information security concerns, like (1) the still not sufficiently solved confidentiality of data at rest over its life-cycle in the cloud, (2) the problem of verifiability of operations and calculations delegated to the cloud, and (3) the threat of the predominant user

privacy disaster in many commercially available cloud services. To address these challenges, a set of **suitable cryptographic primitives** was chosen or proposed by the applicants of the project, and three applications in real-world use cases selected for demonstrating the feasibility of the approach[4]. Several requirements for the implementation of the use cases were and are currently developed in three other deliverables of the project[5]. We will use the paradigm of design patterns to describe and explain the applications and situations for which the cryptographic primitives are intended to provide solutions. We will develop the **cloud security patterns** that are associated with the PRISMACLOUD primitives. This will be strictly done from an end-user perspective, so that the patterns may reveal additional requirements on both the cryptographic primitives themselves, as well as additional requirements on the three demonstrator use cases, even though the scope of this deliverable is to design domain independent security models. On the one hand, the patterns shall provide guidance for the scientists and engineers working on the development and implementation of the cryptographic primitives. The patterns shall help explain the particular needs and requirements of end-users. On the other hand, the patterns shall explain the potential and capabilities of the PRISMACLOUD primitives to end users and other stakeholders beyond the very specialized communities who develop and implement the cryptographic primitives. We intend to bring our PRISMACLOUD patterns into existing collections of cloud security patterns.

In the last chapter we will revisit the cloud computing risks and connected threats and will evaluate, which of the earlier identified and listed risks are covered by the PRISMACLOUD primitives to what extent, and particularly look for threats which are not covered by the PRISMACLOUD primitives and could lead to further development of project results.

[4] This is also reflected in the structure of the consortium: It consists of the research partners associated with the cryptographic primitives, furthermore of partners representing the cloud providers and the end-users associated to the proposed three use cases, and third, the partners in non-technical orthogonal function (for usability issues, legal issues; for standardisation, business deployment, cybercrime research)

[5] End user and HCI (Human Computer Interaction) requirements in D2.1, use case specification in D2.3, generic security objectives and best practices in D3.1

3 Current ontologies and reference architectures

Through the economies of scale, cloud providers are in the position to amass huge processing and storage capabilities in server farms and data centers for a considerably lesser cost and effort than an end customer who sets up a proper ICT infrastructure on its own. This applies to the basic ICT infrastructure (e.g. per CPU or server cost), as well as to the operating personnel, and to all the organisational maintenance and operation details, like the implementation of a dependable storage architecture (including data backup) and the implementation of a continuous lifecycle from procurement to out phasing of equipment.

3.1 Cloud computing definition and principles

According to “The NIST definition of Cloud Computing”, [2]

> *'**Cloud Computing** is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and devices) that can be rapidly provisioned and released with minimal management effort or service provider interaction'.*

Cloud Computing is a specialized distributed computing paradigm, as it differs from traditional distributed computing approaches in that

- it is massively ***scalable***,
- can be encapsulated as an ***abstract*** entity that delivers different levels of services to customers outside the cloud,
- it is driven by economies of scale,
- it offers ***flexibility***; i.e., the services can be dynamically configured (via virtualization or other approaches) and delivered on demand,
- it provides the capability to transfer workload to other engines/data centres, i.e., ***outsourcing***.

Moreover, the cloud computing paradigm lays its foundation on the following three main aspects which are perceived as the real novelties that the cloud landscape is bringing to the world:

1. The illusion of infinite computing resources available on demand.
2. The elimination of an up-front commitment: a company can start small and increase cloud resources only when there is an increase in their needs
3. The ability to pay for use of computing resources on a short-term basis (as needed) and release them (as needed) – Rewarding conservation by letting machines and storage go when they are no longer useful

Those three aspects mixed together virtually eliminate the need to plan far ahead for provisioning your service, leaving the user the capabilities to have its service purchased to grow or decrease with a fast pace, well according to the needs and trends of the actual market landscape.

The cloud paradigm lays its foundation on a certain number of technology enablers which can be identified in the following:

Datacenter. The original data center started as a private server room hosted within the organization's facility containing many individual servers running single applications. In the early days of data centers, most organizations were responsible for maintaining the servers and software, and required a number of personnel resources to manage the servers as well as the facility. A data center (sometimes called a server farm) is a centralized repository for the storage, management, and dissemination of data and information. Typically, a data center is a facility used to house computer systems and associated components, such as telecommunications and storage systems. In the Cloud computing era, the datacenter facilities has evolved from server farms hosting the computing facility dedicated to internal IT infrastructure to massive infrastructure, geographically distributed facilities exposing cloud service to end-users. Moreover datacenter has shifted from the typical three layered architecture to a more flat unified fabric, trying to avoid the bottlenecks related to the east-to-west (aka server to server) traffic.

Virtualization. Virtualization is a technique used to build a virtual resource over an underlying physical infrastructure. Virtualization typical areas are server virtualization (creation of virtual machine of a physical server), Storage virtualization (partition of storage appliance), network virtualization (wide range of technologies starting from legacy VLAN, to switch virtualization and VXLAN protocols), Desktop virtualization and application virtualization. The key benefits of virtualization is the possibility to optimize the usage and the performances of the underlying physical infrastructure, the inherent capability of segregating the virtual resources (which enable a multi-tenancy features) and the possibility to create and remove resources by simple provision/de-provisioning management actions.

High-bandwidth interconnections. Cloud computing lays its foundation on connectivity, hence the relevant aspect of interconnection capabilities. Cloud computing end-users are greed bandwidth consumer, and have been generating over recent years a massive increase of internet traffic. Cloud service providers are closely following recent technology improvements on link speed technology for both WAN interconnections and optical data center fabric areas, and adapting their infrastructure to cope with the growing demands in terms of bandwidth and interconnection capabilities

Automation. The inherent needs of cloud computing framework has led Cloud service providers to evolve their IT infrastructure management, from a rigid structure where the provision and the de-provision of resources has to follow strict operational rules, involving people from different internal departments (network, hosting, application), to a more elastic framework which leverages on automation tools to create resources and make them available as service to the end-user. This automation framework is a software layer which is transversal to all data-center layers spanning from the provisioning of service and the network activities to database and billing operations.

In the following chapter we describe more in detail the actual cloud landscape, by tracing its origins and its predecessor, up to providing current and most typical cloud service and deployment model description.

3.2 Historical account

The original idea of Cloud computing was forecasted back in 1961, by computing pioneer John McCarthy who predicted that "computation may someday be organized as a public utility"— and went on to speculate how this might occur.

We can consider the cloud computing as a culmination of numerous attempts at large scale computing with seamless access to virtually limitless resources, being the most relevant **utility computing**, **grid computing and edge computing**. Most of those framework shares a certain number of aspects with Cloud computing paradigm, being the cloud unique combination of those feature casted over different scenarios.

Grid computing is a further evolution on the idea of compute clusters by creating a distributed virtual supercomputer. Conventional HPC (High Performance Computing) systems are tightly coupled to their other cluster modules through either a high-speed network backbone or switch fabric. Grid computing brings together HPC and distributed computing. In a grid computing environment, a controller system packages portions of the problem-space workload and distributes the pieces to other systems. It is the job of the controller to receive, interpret and package the receipt of individual solution pieces. The idea of grid computing is to utilize a shallow footprint on nodes across a vast computing environment. A widely known example of the grid computing paradigm is the Folding project[6]used to perform protein folding simulation, an embarrassingly parallel yet computationally intense computation. Distributed grid nodes run a background application that utilizes unused CPU and GPU power and accept problem segments in the form of work units from a controller system at Stanford University. Work units are further break-downs of different simulation problems such as folding simulations of Alzheimer's disease or sickle-cell anemia.

Three criteria have been established to formulate the grid computing model:

- Loosely-Coupled. Loosely-coupled systems are made of separate, distinct and autonomous subsystems that have their own resources. Tightly-coupled systems often share system resources such as memory and are connected with a short-distance high-speed network topology or bus.
- Geographically Dispersed. Grid computing nodes communicate through standard networking protocols and are able to take advantage of asynchronism. This means that nodes do not need to be located in close proximity and do not require a constant synchronized communication mechanism.
- Heterogeneous. Heterogeneity is easily exploited in grid computer infrastructures. Most application designs allow for a variety of operating system and node hardware architecture possibilities within the entire grid.

Utility computing is a service provisioning model in which a service provider makes computing resources and infrastructure management available to the customer as needed, and charges them

[6]https://folding.stanford.edu/

for specific usage rather than a flat rate. Like other types of on-demand computing (such as grid computing), the utility model seeks to maximize the efficient use of resources and/or minimize associated costs. Utility is the packaging of computing resources, such as computation, storage and services, as a metered service. This model has the advantage of a low or no initial cost to acquire computer resources; instead, computational resources are essentially rented.

Edge Computing is pushing the frontier of computing applications, data, and services away from centralized nodes to the logical extremes of a network. It enables analytics and knowledge generation to occur at the source of the data. This approach requires leveraging resources that may not be continuously connected to a network such as laptops, smartphones, tablets and sensors. Edge Computing covers a wide range of technologies including wireless sensor networks, mobile data acquisition, mobile signature analysis, cooperative distributed peer-to-peer ad hoc networking and processing also classifiable as Local Cloud/Fog computing and Grid/Mesh Computing, distributed data storage and retrieval, autonomic self-healing networks, virtual cloudlets, remote cloud services, augmented reality, and more.

All the frameworks depicted so far, due to their inherent nature of dispersed systems allowing remote access to end-users, shows the following security concerns:

	Utility computing	**Grid computing**	**Edge computing**
Authentication	high	high	high
Accounting	high	medium	low
Data Privacy	high	medium	high
Data integrity	high	high	high
Tenant isolation	high	medium	medium

Table 1: Security concerns of computing frameworks

Regarding the technology enablers used to implement a security framework, only the Grid Computing has developed a proprietary framework called GSI (Grid Security Infrastructure), while the utility computing and edge computing leverage on more 'legacy' security architectures, which are composed of dedicated HW appliances (Firewall, IPS, IDS, Threat management platforms) and standard cryptography algorithms (X.509 certificates, digital signatures, symmetric encryption).

3.3 Cloud computing ontologies and taxonomies

A common definition of ontology applied to computer science area reports the following:

"*In computer science and information science, an ontology is a formal naming and definition of the types, properties, and interrelationships of the entities that really or fundamentally exist for a particular domain of discourse. It is thus a practical application of philosophical ontology, with a taxonomy*."

In recent years, since the broad adoption and deployment of cloud paradigm, many attempts have been tried to provide a comprehensive ontology/taxonomy to provide an overall model for cloud services, from both operational and business perspective.

The main difficulty to provide a stable and comprehensive framework to describe the cloud paradigm also comes from the fact that the cloud itself has seen a rapid evolution over the years, shifting from a much narrowed focused approach and deployment to a very broad landscape, enriched with wide range of services, deployment models and business perspectives.

In the following paragraph we will present the most important taxonomy models, taking into account that we can still consider NIST model as the model with largest consensus, while the other models reflects a wider approach to the cloud problem even though their adoption still do not receive unanimous consensus.

The U.S. National Institute of Standards and Technology, which is currently one of the leading normative powers in cloud computing, provided one of the first attempts for the classification of the cloud paradigm. It defines four deployment models and four service models for cloud computing [2]. "The various cloud deployment models in the NIST cloud definition have implications for the locations of consumer-controlled security perimeters and hence for the level of control that consumers can exercise over resources that they entrust to a cloud." [3] NIST categorizes cloud services in the following three service models:

- Software as a Service (SaaS)

  The user can use the applications from the cloud service provider via a thin client such as a web browser or a program interface. The underlying cloud infrastructure, network, servers, operating system, storage or even individual application capabilities are thereby not managed by the user. [4]

  As the cloud service provider manages most of the parts, which are necessary for providing SaaS, they must also assume more responsibilities. One of the most import responsibilities for the service provider is thereby that all the data remains secure. This is completely new for some software vendors as the customer was responsible for that when he/she worked with an on premise solution. The service provider is further responsible for making their software more secure against security threats, that the software works within the security infrastructure of the customer and that their service defends itself against internal and external threats while operating. [7]

[7] http://cloudstrategies.biz/requirements-for-building-enterprise-saas-applications/

- Platform as a Service (PaaS)

The user can deploy applications. The underlying cloud infrastructure, network, server operating system, storage are thereby not managed by the user but he can at least control the deployed applications and the environment where the applications are hosted. [4]

A PaaS provider is therefore responsible for the operating system, Middleware, Runtime and even Database[8].

- **Infrastructure as a Service (IaaS)**
- The user can control fundamental computing resources such as storage and some networking components (e.g. firewall) including operating systems and applications. The underlying cloud infrastructure is not managed by the user. [4]

The IaaS provider has the least responsibilities compared with PaaS and SaaS provider. The IaaS provider is responsible for managing the physical resources which means that he/she is responsible for the network, servers and clustered machines.

Further service modes are currently being proposed, as e.g. “Network as a Service”, which denotes the provision of an efficient virtual network service to tenants who want to deploy custom routing and multicast protocols in the efficient network infrastructure of a cloud provider [5].

Sam Johnston created a taxonomy, which consists of 6 layers. These layers are:

- **Clients** This Layer represents computer hardware and/or computer software.
- **Services** This layer enables a machine-to-machine interaction over the network.
- **Application** This layer enables the user to run the applications he/she needs in the cloud and therefore he/she does not need to install and maintain applications on his/her local machine.
- **Platform** This layer is equivalent to Platform as a Service
- **Storage** This layer is equivalent to Data Storage as a Service.
- **Infrastructure** This layer is equivalent to Infrastructure as a service[9].
-

The taxonomy of Dave Linthicum consists of 10 layers and they are defined as follows:

- **Storage as a Service** The storage is physically located somewhere in a data centre but it is logically a local storage for applications.
- **Database as a Service** The database is hosted remotely and it can be shared with many other users but it functions like it would operate locally. This enables customers to use

[8] https://www.simple-talk.com/cloud/development/comparing-iaas-and-paas-a-developer%E2%80%99s-perspective/
[9] http://samj.net/2008/09/17/taxonomy-the-6-layer-cloud-computing-stack/

database models which would have cost thousands of dollars if they would operate on premise.

- **Information as a Service** The user can access information that is stored remotely via defined interfaces.
- **Process as a Service** Provides a remote resource that binds other resources together in order to create business processes. An advantage of this service is that it is more agile since processes are easier to exchange than applications.
- **Application as a Service** This layer is equivalent to software as a service. Any application that can be accessed with the browser for example *GoogleDocs.*
- **Platform as a Service** A platform that provides application development, interface development, database development, storage and testing for the customers.
- **Integration as a Service** Provides a complete integration stack including interfacing with applications, semantic mediation, flow control and integration design.
- **Security as a Service** Core security services are delivered over the web.
- **Management/governance as a Service** The user is able to manage cloud services on-demand. Services such as resource utilization, virtualization and uptime management, are meant.
- **Testing as a service** Services which are able to test other cloud applications, websites, and internal enterprise systems.[10]

Intel identified six primary categories, including the three categories from the NIST categorisation. The additional categories are the followings:

- **Service as a Service** This category consists of horizontal services which are used as components within other cloud services such as PaaS, IaaS and SaaS. A typical example of a Service as a Service is *Security as a Service*.
- **Cloud software** Cloud software is software that is used to build and run cloud services.
- **Cloud client** The cloud client contains client-centric services, runtimes and runtime optimizations. [6]

Another example of how cloud computing services can be categorized is provided by C.N. Höfer and G. Karagiannis in *Cloud computing services: taxonomy and comparison*. They categorize all the service with a tree. The example below shows how the tree would look like for *Amazon EC2*.

[10] http://davidlinthicum.sys-con.com/node/811519

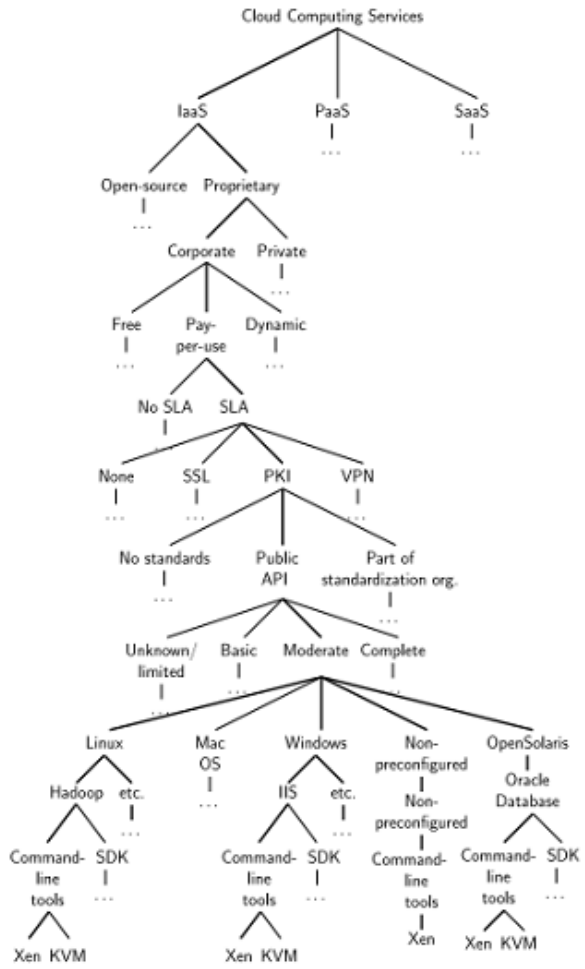

Figure 1: How *Amazon EC2* can be categorized with a tree (cf. [7])

Another existing taxonomy is Youseff's 5-Layer Ontology which is structured as follows:

- **Cloud Application Layer** (similar to SaaS): The cloud application layer is the most visible layer to the end-users of the cloud. Normally, the users access the services provided by this layer through web-portals, and are sometimes required to pay fees to use them)
- **Cloud Software Environment Layer** (similar to PaaS): The second layer in our proposed cloud ontology is the cloud software environment layer (also dubbed the software platform layer). The users of this layer are cloud applications' developers, implementing their applications for and deploying them on the cloud
- **Cloud Software Infrastructure Layer**: The cloud software infrastructure layer provides fundamental resources to other higher-level layers, which in turn can be used to construct new cloud software environments or cloud applications. Which can be further divided into:
    - Computational Resources (IaaS): Services that belong to computational resources are similar to the ones that belong to IaaS as described in the NIST definition.
    - Data Storage as a Service: (DaaS)

- The user can store his/her data remotely and is able to access the data from different devices. These data storage systems have to store the data in a way so that data consistency, reliability, high availability and so on, is ensured. As some requirements are contradicting, DaaS providers simply decided to concentrate on one requirement which is defined in the SLAs. An example of these DaaS-systems is Amazon's S3.
- Communication as a Service (CaaS): As cloud systems need to provide some possibilities for the customers to communicate the concept of Communication as a Servicee merged. This model is the least discussed cloud service but one recent example that belongs to CaaS is Microsoft Connceted Service Framework.

- **Software Kernel** (This cloud layer provides the basic software management for the physical servers that compose the cloud)
- **Hardware and Firmware** (The bottom layer of the cloud stack in our proposed ontology is the actual physical hardware and switches that form the backbone of the cloud.)
  - **Hardware as a Service (HaaS):**
  - The users of HaaS are usually big companies that want to rent physical hardware in order to sublease it to their consumers. The provider is thereby responsible for maintaining the hardware. [8]

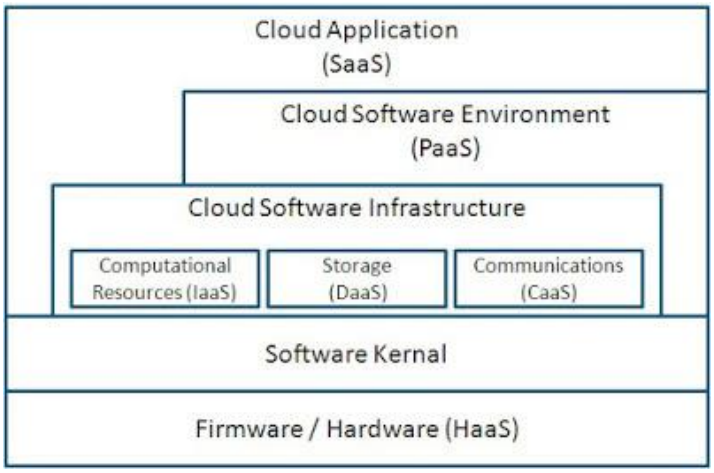

Figure 2: Cloud Ontology as proposed by [8]

A further extension to the work around cloud ontology definition is provided via the concept of XaaS (Everything as a Service) which foresee an even wider range of categories/services which inherits all cloud layers seen so far with even a more detailed granularity. The most prominent of those new categories can be summarized in the following:

- IaaS (Infrastructure As A Service)
- PaaS (Platform As A Service)

- SaaS (Software As A Service)
- CaaS (Communication As A Service)
- MaaS (Monitoring As A Service)

3.4 Reference architectures

NIST [9] has proposed a reference definition of architecture for Cloud computing which, even if it comes from survey operated during 2010/2011, still sounds valid nowadays.

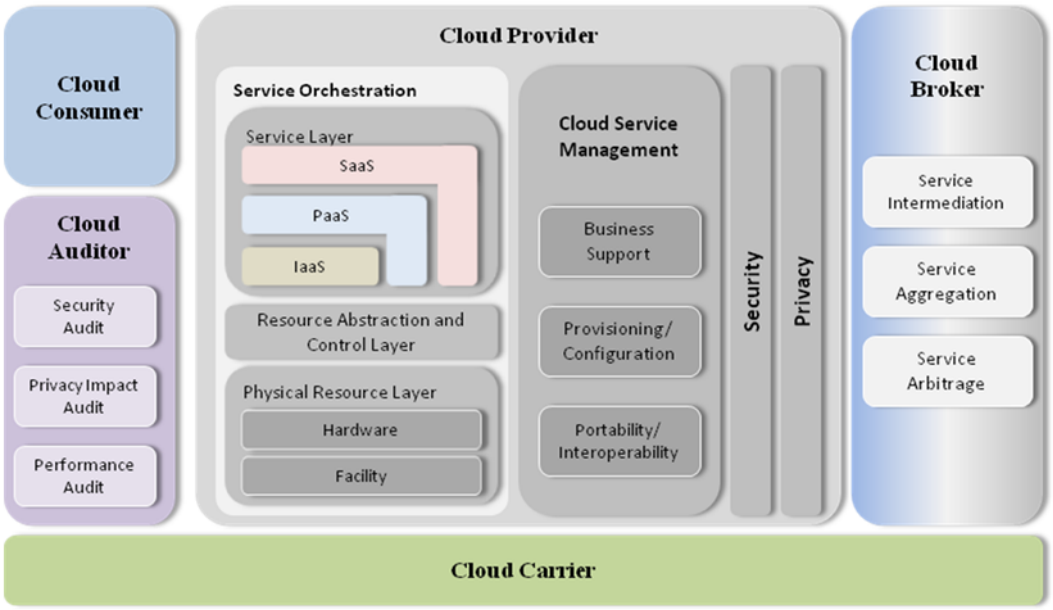

Figure 3: Cloud computing reference model [9]

As depicted above the cloud model adopted by NIST foresee the following five actors/roles

- **Cloud Consumer**, A person or organization that maintains a business relationship with, and uses service from, Cloud Providers.
- **Cloud Provider**, A person, organization, or entity responsible for making a service available to interested parties. Cloud Auditor A party that can conduct independent assessment of cloud services, information system operations, performance and security of the cloud implementation.
- **Cloud Broker**, An entity that manages the use, performance and delivery of cloud services, and negotiates relationships between Cloud Providers and Cloud Consumers.
- **Cloud Carrier**, An intermediary that provides connectivity and transport of cloud services from Cloud Providers to Cloud Consumers.
- **Cloud Auditor**, A party that can conduct independent examination of Cloud Service controls with the intent to express an opinion thereon. Audits are performed to verify conformance to standards through review of objective evidence.

This landscape of actors/roles, joined with the service models (IaaS, SaaS, PaaS) provides a huge landscape with wide deployability capability over the following reference deployment scenarios:

I. **Private cloud**: The cloud infrastructure is provisioned for exclusive use by a single legal organization comprising multiple consumers (e.g., business units). It may be owned, managed, and operated by the organization, a third party, or some combination of them, and it may exist on or off premises. Private cloud deployment, w.r.t a public deployment, can provide cloud services with enhanced security features as dictated by user requirements, higher availability as resources are dedicated only to a certain group of users, and higher flexibility due to the fact that services can be tailored as per specific user requirements.

II. **Community cloud**: The cloud infrastructure is provisioned for exclusive use by a specific community of consumers from organizations that have shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be owned, managed, and operated by one or more of the organizations in the community, a third party, or some combination of them, and it may exist on or off premises.

III. **Public cloud**: The cloud infrastructure is provisioned for open use by the general public. It may be owned, managed, and operated by a business, academic, or government organization, or some combination of them. It exists on the premises of the cloud provider.

IV. **Hybrid cloud:** The cloud infrastructure is a composition of two or more distinct cloud infrastructures (private, community, or public) that remain unique entities, but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting for load balancing between clouds).

Moreover, the relationships among different cloud operators, considered as different administrative entities, are ruled by the following definitions:

The **Single Cloud Model** is the most typical model, followed by big cloud operators (i.e., Google, Amazon) who deploy different data centre infrastructures in geographical diversity, with connectivity on Tier-1 and Tier-2 domains. Connectivity among different data centres is provided based on economic agreements with ISPs, but no agreements with other cloud operators are established.

The **Federated Cloud model** is the model in which smaller cloud operators (with connectivity on Tier-2, and Tier-3 domains) join together to form a federation (a sort of super-cloud entity) in order to achieve economics improvements in terms of increasing capacity to serve more end-users and enlarge the platform of offered services. One major advantage of the federated cloud model is the possibility for smaller cloud operators to use resources located in data centres that belong to other cloud operators who have joined the federation. From user perspective, the differences between the cloud operators joining the federation are transparent. One of the major concerns of this model related to the protocols used to establish the federation as well as issues about user identity, security and privacy.

The **Interconnected cloud model** is the last model of interaction among cloud operators. It's similar to the previous model except for the fact that no federation is formed. The Interconnected Cloud Model foresees that each cloud operator maintain its administrative role, while also establishes

economic agreements with other partners to achieve service mobility, i.e., offload its computing and hosting capacity, with the final aim to ensure proper QoE to its own customers.

3.5 Major cloud service providers

More and more companies are using cloud services and therefore the cloud infrastructure sales, account for over 25% which is almost $6.3 billion in the first quarter of 2015.[11]A forecast regarding the growth of cloud-based platforms states that more than 60% of all companies will have at least 50% of their infrastructure in a cloud. [10]

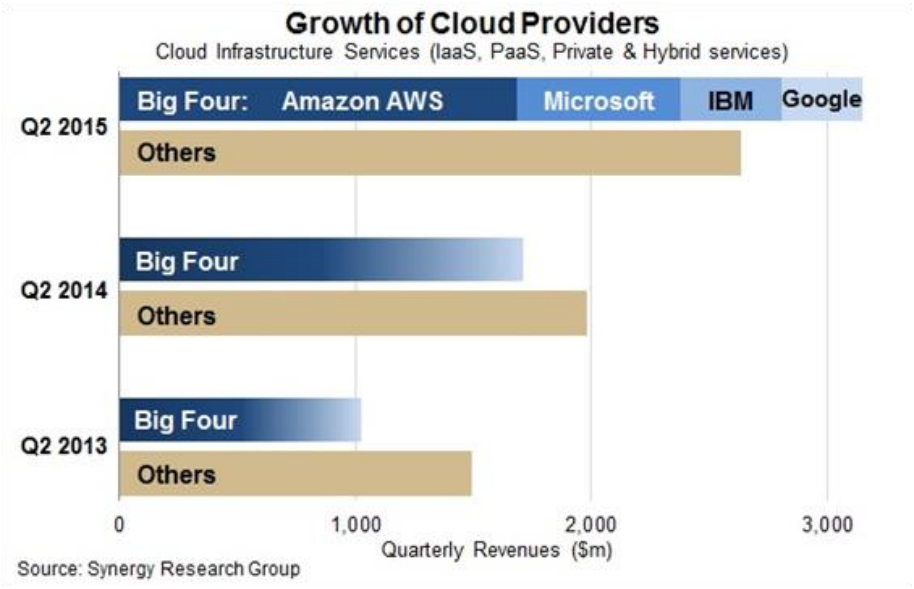

Figure 4: Growth of Cloud Providers

The Computerworld Forecast Study 2015 figured out that cloud computing projects are the most important projects in 16% of the IT departments that they surveyed, followed by legacy systems modernization/replacement (12%). Further, 42% of the companies said that they will increase their spending on cloud computing.[12]

[11] http://cloudtimes.org/2015/07/10/enterprises-migrating-it-infrastructure-investments-to-the-cloud-idc-report/

[12] http://www.forbes.com/sites/louiscolumbus/2014/11/26/computerworlds-2015-forecast-predicts-security-cloud-computing-and-analytics-will-lead-it-spending/

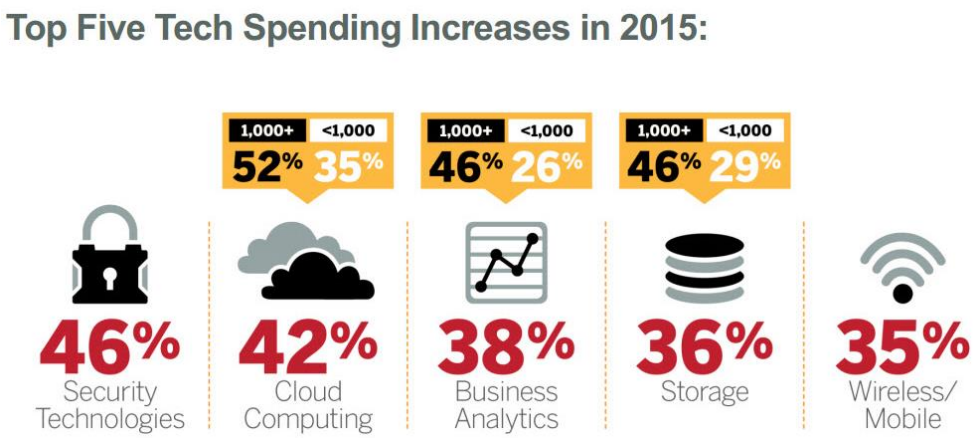

Figure 5: Tech Spending Increases 2015 cf[12]

The leader in this market is by far Amazon followed by Microsoft and all the other cloud service providers lie far behind of these two.[13]In the following paragraphs we will give a short overview of the major cloud service providers.

3.5.1 Amazon Web Services

In the year 2006 Amazon started with its Simple Storage Service (S3) and the Elastic Compute Cloud (EC2). One point which made EC2 so successful was that Amazon did not restrict what developers can do and so they can use their virtual machine as their local computer.[14]This means you can have as many virtual servers as you want and you can configure security and networking and manage storage as you want.[15] Amazon is located at many places all over the world and this means that they have a diverse customer base. AWS has over 10 times more cloud IaaS compute capacity than the next 14 largest cloud service providers together. This made it possible to attract many technology partners which further enable AWS to be very innovative, agile and responsive to the market. It expands its offers very quickly and has therefore the biggest spectrum of IaaS features and PaaS-like capabilities.[16]

3.5.2 Microsoft Azure

Microsoft entered the cloud market in the year 2008 with Microsoft Azure. [11] Microsoft Azure consists of Windows Azure, SQL Azure and Azure AppFabric. Windows Azure provides scalable storage space and on-demand computation of cloud applications. SQL Azure provides a database with additional capabilities in comparison to a normal SQL Server. Azure AppFabric simply provides a set of .NET Services. [12] Microsoft Azure is a PaaS provider and is the most important part of

[13]http://www.webopedia.com/Blog/cloud-computing-market-leaders-2015.html
[14]http://www.wired.com/2012/11/amazon-3/
[15]http://www.enterprisetech.com/2014/11/14/rare-peek-massive-scale-aws/
[16]http://www.gartner.com/technology/reprints.do?id=1-2G45TQU&ct=150519&st=sb

Microsoft's hybrid cloud (Cloud OS). Cloud OS combines Microsoft Azure, Windows Server, Microsoft System Center and the public cloud. Furthermore, Microsoft Azure can be accessed from every end-user device as it does not rely on the underlying hardware configuration.[17]

3.5.3 IBM SmartCloud

IBMs SmartCloud includes infrastructure as a service, platform as a service and software as a service solutions. All these different kinds of services can be offered through a public, private or hybrid cloud. These offers are coffered by the following three solutions: SmartCloud Foundation, SmartCloud Services and SmartCloud Solutions [13]. 2007 was the year when IBM started to develop a cloud strategy. The goal of their cloud computing strategy was to serve enterprise customers and to close the gaps of existing cloud environments. IBM teamed up with Google, in the same year in order to distribute information about cloud computing at universities. Four years later the IBM cloud solution named SmartCloud began to grow in a steady manner. In the same year IBM announced that already 400 companies use their cloud solution.[18]

3.5.4 Google Cloud Platform

Google released their Google App Engine (GAE) in the year 2008.[19]Google App Engine is a Platform as a Service solution and enables the user to develop applications. All applications are sandboxed and can therefore run in a safe environment.[20] A real competitor to Amazons S3 cloud storage is the Google Cloud Storage as Google also offers their customers to store their data on Google's infrastructure. [21] After the release of Google Cloud Storage, Google introduced the Google Cloud Platform which consists of many different cloud services. These services are thereby split up in these four categories: Compute, Storage, Big Data, Services and all cloud services are assigned to the categories as shown in figure 4-2.[22] For deploying large scale software it is important to consider that a manual configuration will result in a maintenance challenge. For customers, who use Google App Engine that is already managed by GAE but for the others, developed Google, the Google Cloud Deployment Manager which makes designing, sharing, deploying and managing complex cloud solutions easier.[23]

[17] http://marketrealist.com/2014/08/microsoft-appears-path-future-success/
[18] http://cloud-computing-in-the-cloud.com/ibm-cloud-computing-wikipedia-article/59
[19] http://googleappengine.blogspot.nl/2008/04/introducing-google-app-engine-our-new.html
[20] https://cloud.google.com/appengine/docs/python/modules/
[21] https://cloud.google.com/storage/docs/overview
[22] https://cloud.google.com/
[23] http://googlecloudplatform.blogspot.nl/2014/03/bringing-together-best-of-paas-and-iaas.html

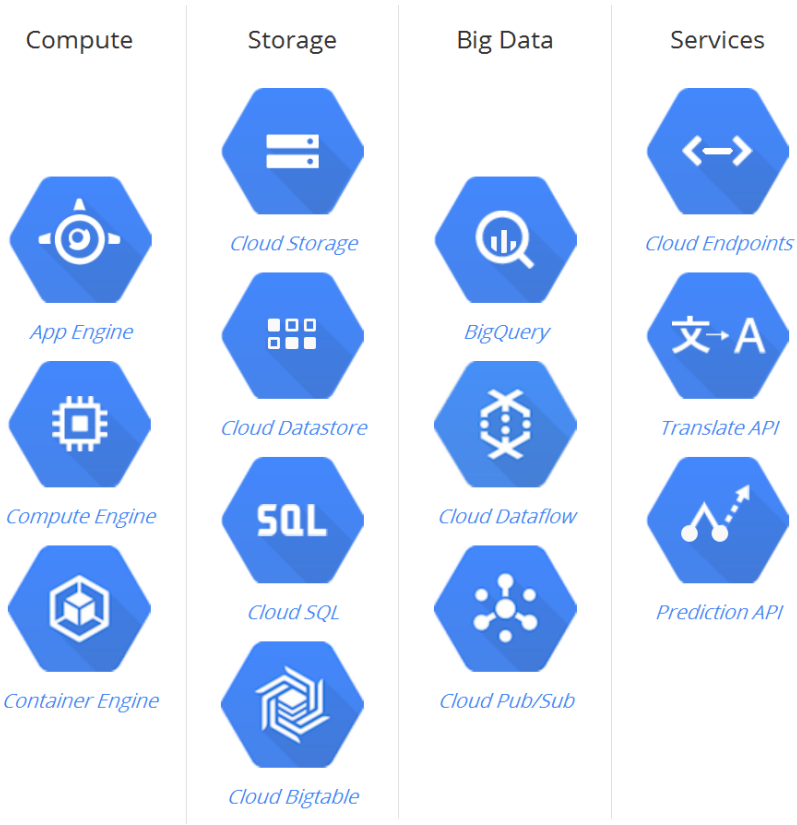

Figure 6: Taxonomy of Google Cloud Platform cf.[19]

3.5.5 Interoute Virtual Data Center VDC

Interoute is the owner-operator of one of Europe's largest networks and a global cloud service platform which encompasses 14 data centres and 31 colocation centres, with connections to 195 additional third-party data centres across Europe. Its full-service unified ICT platform serves international enterprises and many of the world's leading service providers, as well as governments and universities.

The Interoute pan-European infrastructure is designed for the delivery of enterprise IaaS and virtualized services and is directly interconnected through a network owned by Interoute.

State-of art IRT product is VDC2.0, a multi-tenant Infrastructure as a Service (IaaS) platform for on-demand computing and cloud hosting with integrated applications that enables either private or public cloud computing, offering public simplicity with private cloud security.

Customers build their own virtual data centre through a VDC Control Centre graphical interface: they can provision virtualized servers, storage volumes and network segments within a specific zone (i.e., an Interoute site, as London, Amsterdam, Paris, Madrid, Geneva, Munich, Stockholm, and Berlin). Into a VDC, customers can deploy as many Virtual Machines and appliances (e.g. firewalls, IDS/IPS, IP PBXs) as they desire, grouped into individual and separated network domains (i.e. Layer2 LANs).

The VDC service is built by the following main logical components:

- Cloud orchestration: provides the composition of system components to support the arrangement, coordination, and management of infrastructure resources (computation, storage and network) in order to provide cloud services to the customers.
- Compute: provides CPU and memory resources to be virtualized for multi-tenancy purposes through the use of hypervisors.
- Storage: provides shared disk resources for hosting customers' virtual appliances as well as dedicated storage for the appliances themselves.
- Networking: provides pre-provisioned high performance, multi-tenanted network connectivity between all the VDC components, up to the customer LANs (internal and external) and the Internet.

To satisfy customers' information security demands, VDC service is built to assure confidentiality and integrity of valuable private data. Confidentiality and integrity are addressed at the network layer by isolating and separating network traffic, maintaining it only within the scope of the owning customer organisation. VDC service is fully integrated with Interoute MPLS/IP network.

3.6 Cloud service providers' security options

In the following, we present the result of an investigation of some cloud services, regarding security and privacy concerns. These cloud service providers have been chosen as they are partially the major cloud service providers and additionally some smaller providers in order to have IaaS, PaaS and SaaS providers in the table. The categorization of the cloud services was thereby obtained from the website: *http://cloudtaxonomy.opencrowd.com/*. The reason, why this table was added, is to get a good overview about the current status of security and privacy guarantees that cloud service provider offer.

	Privacy policy	Privacy guarantees	To whom belongs the data	Which rights are granted to whom
Amazon S3	Applies to the customer's account information* but not to the content that customers store there.	*Amazon* does not disclose, move, access or use customer content except as provided in the customer's agreement with *AWS*.[24]	The customer can specify the *AWS* regions in which he/she wants to store his/her content. *Amazon* will not move the data without notifying the customer, except it is necessary to comply with the law.	The customer is responsible for all activities that are made with his/her account and *Amazon* has no rights to access the content that customers store there, except the customer agrees to it.[25]
Rackspace Mosso Cloud [14]	Applies to personal information and other information from or about visitors of the website, customers using *Rackspace* services, users of any mobile-device applications, service providers, business partners, job applicants and other third parties that interact with *Rackspace*.	Personal information gathered by *Rackspace* will only be used internally and not revealed to anybody outside, except the customer agrees to this disclosure. If *Rackspace* needs to reveal customer information to a third party, they only do it in a reasonable extent or as permitted by law.	*Rackspace* collects information, they receive from other sources (social media platforms), information that customers give them and they collect information of their customers and their customers devices automated.	The customers have the right to access, correct and request the deletion of their own personal information in accordance with the law. The customers can also oppose some data processing practices or revoke consent previously granted.
webMethods AgileApps Cloud (formerly AgileApps Live)[26]	Applies to personal information, services information, end user information and registration information.	Services and End User Information are used to provide cloud services, to ensure a satisfying performance, to maintain, fix and upgrade the system. Furthermore the information is used to better		They may access information only for providing services, for preventing or addressing technical problems, for customer support reasons or because it is required by law.

[24] https://aws.amazon.com/privacy/?nc1=h_ls
[25] https://aws.amazon.com/agreement/
[26] http://www.softwareag.com/corporate/privacy_agileapps.asp
* Name, username, phone number,e-mail address, billing data in connection with the AWS- account of a customer

		serve the customer and also if it is required by law.		
Ping Identity [27]	Applies to all people who provide personal information (customer, job applicant, etc.), all locations where *Ping Identity* operates and all methods of contact (website, services, etc.)	*Ping Identity* uses personal information of the customers for providing requested products and services, inform about other products they offer and to manage their sites and services. Moreover, the information may be shared to perform services, enrol customers to communications, make visits on the website more personalized and so on. If a third party is needed to perform some services the customer information might be shared also with them.	*Ping identity* has the control over the personal data and has policies and rules in place so that an unauthorized access is not possible.	*Ping identity* ensures that the personal information is maintained accurately. The customer can view, make corrections or modify information at any time or decide that *Ping identity* is not allowed to use customer information to provide services**.**
Google [28]	Applies to all services that Google Inc. and their affiliates offer, including Google services which are provided on Android devices and services provided on other sites.	*Google* uses all information that they get from their services to provide, maintain, protect and improve their services. Furthermore, the information is also used to develop new services, to protect *Google* and their users as well as for providing customized content to their users.	*Google* collects information that the customers give them when they create a Google Account or that they get from their services when the customers use them. This information includes device information, log information, location information, unique application numbers, local storage, cookies and similar technologies.	The users can review and update their *Google* activity controls, review and control all the information regarding their *Google Account*, view and edit their preferences regarding *Google Ads* and define what other people can see of their profile. Additionally are users able to control with whom they share their data, delete information from their *Google*

[27] https://www.pingidentity.com/en/legal/privacy.html
[28] http://www.google.com/intl/en/policies/privacy/#infochoices

				Account and choose whether their profile name and profile picture appear in ads.
Microsoft Azure	Applies to all *Microsoft* online services and the offers connected to them.[29]	*Microsoft* uses the information of their customers only to be able to provide them the services they want but they will neither use the data of their customers in general nor for advertising.	The customers control the collection, use and distribution of their data. *Microsoft* thereby tries to be transparent with their privacy practices and responsible in managing the data that they store and process.	The data that is hosted on *Azure* still belongs to the customer. The customer has the control over where the data is stored, how it is processed and deleted. *Microsoft* releases specific data only when it is requested by court or another legal authority and to the government if they have a court order for the content or a subpoena for account information.[30]

Table 2: Security and privacy concerns analysis for selected cloud services

Summarizing the table above, one can say that all companies use personal customer data for providing services or if data is requested by the court. One fact where Ping Identity differs, compared with the others, is that they have the control over the personal information of the customer while the other providers state that the customer keeps the control of their data.

[29] https://www.microsoft.com/privacystatement/de-DE/OnlineServices/Default.aspx
[30] https://azure.microsoft.com/en-us/support/trust-center/privacy/

3.7 Recommendations for Cloud security framework

NIST provides a reference scheme for a cloud security architecture, which represents sort of a high level recommendation, identifying macro-blocks of functional areas super-imposed to the actual NIST cloud taxonomy [15].

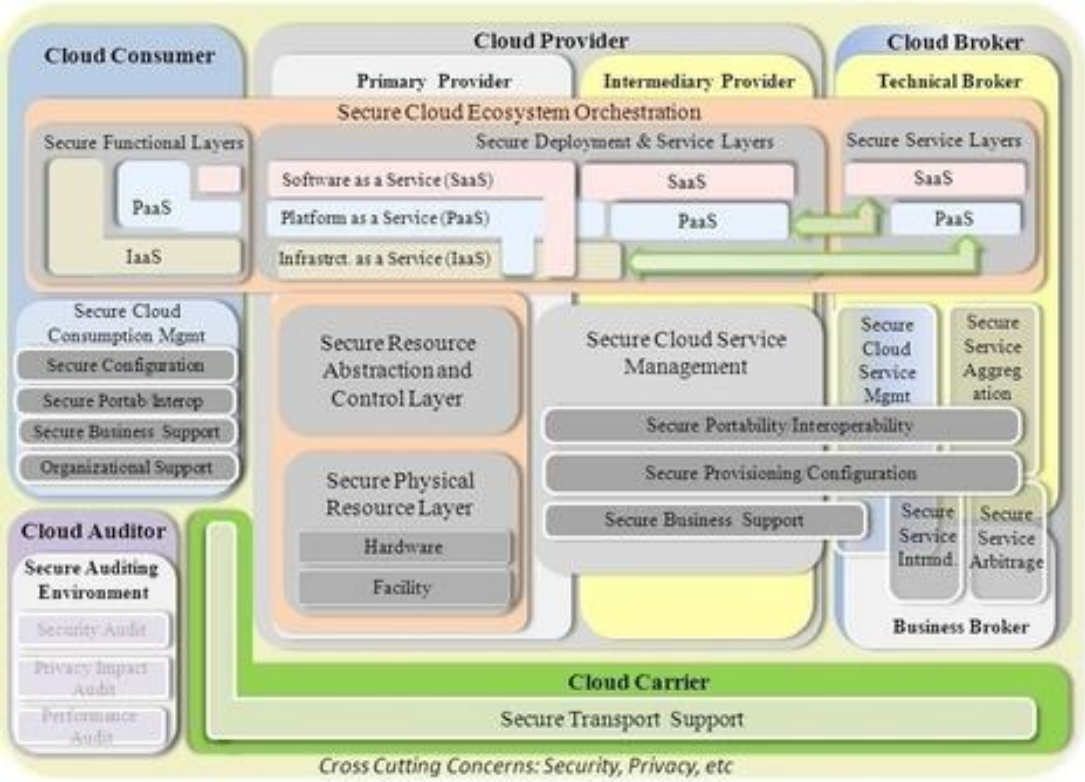

Figure 7: NIST high-level framework for cloud security architecture

The reference schema provided only identifies functional areas without providing any technical details regarding the implementation framework to be used which is left to the Cloud Provider. The reference schema identifies the following key functional areas (identified on a per cloud actor basis):

- Cloud Consumer
  - Secure Cloud Consumption Management which includes all of the functions that are necessary for the management and operations of the service used by the cloud Consumers;
  - Secure Configuration which includes any capabilities, tools, or policies that ensure the secure configuration of cloud resources and compliance with the applicable security standards, specifications, and mandate;
  - Secure Portability/Interoperability which ensures that data and applications can be moved securely to multiple cloud Ecosystems while the risk mitigation measures in place are commensurate with the data security and privacy requirements and necessary level of protection.

- Secure Business Support which includes capabilities such as identity provisioning and credential management to the organization's employees and contractors through access control policies, business continuity plans, and various productivity tracking mechanisms for use by the Consumer;
- Secure Cloud Ecosystem Orchestration (Functional layer) which is the set of Security Components a cloud Consumer implements to secure the cloud Functional Layer depends upon the particular cloud service model used.

- Cloud Provider
  - Secure Cloud Ecosystem Orchestration
    - Secure Deployment & Service Layers which is the set of Security Components a cloud Provider can implement to secure the Service Layer depends upon the particular type of cloud service offered;
    - Secure Resource Abstraction and Control Layer (Hardware & Facility) which is Layer is the Architectural Component that contains the Security Components a cloud Provider would implement to provide and manage secure access to its physical computing resources through software abstraction;
    - Secure Physical Resource Layer (Hardware & Facility) which is an architectural subcomponent that contains the Security Components needed to secure physical computing resources.
  - Secure Cloud Service Management
    - Secure Provisioning and Configuration which includes all Security Components (such as capabilities, tools, or policies) that ensure the secure configuration and provisioning of cloud resources, with particular focus on compliance with the applicable security standards, specifications, and regulations;
    - Secure Portability and Interoperability which ensures that data and applications can be moved securely to multiple cloud Ecosystems, as established by cloud Consumer security requirements;
    - Secure Business Support which entails the set of business-related services dealing with the Provider's Customers and supporting security processes.
- Cloud Broker
  - Secure Service Aggregation: This Architectural Component includes the Security Components that support the fusion and integration of multiple isolated services into one or more new services. The cloud Broker provides data integration and ensures the secure movement of data between the cloud Consumer and multiple cloud Providers based upon the security policies of the Consumer.
  - Secure Service Arbitrage: This Architectural Component is similar to the Secure Service Aggregation component, except that the services being aggregated are not fixed. Service arbitrage means a cloud Broker has the flexibility to choose services from multiple Providers. The cloud Broker can, for example, use a credit-scoring service to measure and select a cloud Provider with the best score.
  - Secure Service Intermediation: This Architectural Component includes the Security Components that facilitate the enhancement of a given service by allowing the cloud

Broker to improve some specific capability and offer value-added services to cloud Consumers, while ensuring the security policies of the Consumer are maintained. The improvement can be managing access to cloud services, identity management, performance reporting, enhanced security, etc.

- Secure Cloud Service Management: This Architectural Component includes all Security Components that support the management of all service-related functions (technical and business) that are necessary for the operations of the services offered by the cloud Broker. Secure Cloud Service Management can be described from the perspective of the:
  - business support requirements;
  - provisioning and configuration business requirements;
  - portability and interoperability business requirements.
- Secure Cloud Ecosystem Orchestration: This Architectural Component includes all Security Components that a Technical Broker needs to implement to secure the functionality implemented and the additional services offered based upon the cloud deployment model (e.g., Private, Public) and service model (SaaS, PaaS, or IaaS).

In the reference model provided by NIST, the additional role of Cloud Broker (an entity that manages the use, performance and delivery of cloud services, and negotiates relationships between cloud Providers and cloud Consumers) is foreseen. This additional role, whilst not taking part in the cloud taxonomy casted over PRISMACLOUD framework, can be of the utmost importance during the subsequent analysis for the exploitation activities.

In the following table we present a set of recommendations (taken from NIST) to be used as guidelines to drive the Cloud Consumer when choosing to move data/infrastructure over the cloud.

Areas	**Recommendations**
Governance	Extend organizational practices pertaining to the policies, procedures, and standards used for application development and service provisioning in the cloud, as well as the design, implementation, testing, use, and monitoring of deployed or engaged services. Put in place audit mechanisms and tools to ensure organizational practices are followed throughout the system lifecycle.
Compliance	Understand the various types of laws and regulations that impose security and privacy obligations on the organization and potentially impact cloud computing initiatives, particularly those involving data location, privacy and security controls, records management, and electronic discovery requirements. Review and assess the cloud provider's offerings with respect to the organizational requirements to be met and ensure that the contract terms adequately meet the requirements. Ensure that the cloud provider's electronic discovery capabilities and processes do not compromise the privacy or security of data and applications.
Trust	Ensure that service arrangements have sufficient means to allow visibility into the security and privacy controls and processes employed by the cloud provider, and their performance over time. Establish clear, exclusive ownership rights over data. Institute a risk management program that is flexible enough to adapt to the constantly evolving and shifting risk landscape for the lifecycle of the system. Continuously monitor the security state of the information system to support ongoing risk management decisions.
Architecture	Understand the underlying technologies that the cloud provider uses to provision services, including the implications that the technical controls involved have on the security and privacy of the system, over the full system lifecycle and across all system components.
Identity and Access Management	Ensure that adequate safeguards are in place to secure authentication, authorization, and other identity and access management functions, and are suitable for the organization.
Software Isolation	Understand virtualization and other logical isolation techniques that the cloud provider employs in its multi-tenant software architecture, and assess the risks involved for the organization.

Data Protection	Evaluate the suitability of the cloud provider's data management solutions for the organizational data concerned and the ability to control access to data, to secure data while at rest, in transit, and in use, and to sanitize data. Take into consideration the risk of collating organizational data with that of other organizations whose threat profiles are high or whose data collectively represent significant concentrated value. Fully understand and weigh the risks involved in cryptographic key management with the facilities available in the cloud environment and the processes established by the cloud provider.
Availability	Understand the contract provisions and procedures for availability, data backup and recovery, and disaster recovery, and ensure that they meet the organization's continuity and contingency planning requirements. Ensure that during an intermediate or prolonged disruption or a serious disaster, critical operations can be immediately resumed, and that all operations can be eventually reinstituted in a timely and organized manner.
Incident Response	Understand the contract provisions and procedures for incident response and ensure that they meet the requirements of the organization. Ensure that the cloud provider has a transparent response process in place and sufficient mechanisms to share information during and after an incident. Ensure that the organization can respond to incidents in a coordinated fashion with the cloud provider in accordance with their respective roles and responsibilities for the computing environment.
Accountability	Cloud provider should ensure that suitable chain of accountability is in place to ensure to members of a cloud ecosystem that obligations to protect data are observed by all who process the data, irrespective of where that processing occurs. This not only applies when a data subject directly uses cloud services but also when services are provided in an enterprise loud setting.
Transparency	Ensure that cloud provider is able to expose details related to transparency in relation to how data are managed, where are stored and when retrieved after potential loss. Cloud Provider should also be ready to formally provide audit results to end-user. Transparency must be provide via a proper and comprehensive technical and operational framework by the Cloud provider and regulated via proper SLA definition.

Table 3: NIST recommendation for Cloud Consumers

3.8 Elements of a security aware taxonomy

3.8.1 Introduction

While the previous section is focused on defining a detailed taxonomy of the different cloud services available in the market, this section describes in detail the security functionalities available. Between the broad range of privacy parameters and security controls available to take into consideration, this analysis has been especially focused on the following: Data Security and Storage and Identity and Access Manager.

3.8.1.1 Data Security and Storage

This section details the factors taken into consideration for the security controls or functionalities associated with protecting the Data. Nevertheless the protection of the data is an abstract concept and because the Data can be in different stages during its life cycle within the cloud some concepts have to be defined. Therefore in this study the Consortium has taken into consideration the stages that the data will pass through:

- Data-in-transit: encrypted during transfer to and from a cloud provider,
- Data-at-rest: be encrypted if using simple storage
- Data lineage: Following the path of data (mapping application data flows or data path visualization) is known as data lineage, and it is important for an auditor's assurance (internal, external, and regulatory).
- Data provenance: Provenance means not only that the data has integrity, but also that it is computationally accurate and it is also about documenting/being able to proof accuracy; that is, the data was accurately calculated.
- Data remanence: is the residual representation of data that has been in some way nominally erased or removed.

3.8.1.2 Identity and Access Manager:

The Identity and Access Manger is based on the following pillars:

- Authentication: Authentication is the process of verifying the **identity** of a user or system.
- Authorization: Authorization is the process of determining the **privileges** the user or system is entitled to once the identity is established
- Auditing: In the context of IAM, auditing entails the process of review and examination of authentication, authorization records, and activities to determine the adequacy of IAM system controls, to verify compliance with established security policies and procedures(e.g., separation of duties), to detect breaches in security services (e.g., privilege escalation), and to recommend any changes that are indicated for countermeasures.

The services associated with these pillars are: User management, Authentication management, Authorization management, Access management and Monitoring and auditing.

However this study has also taken into consideration other aspects associated with the Identity and Access Management such as the operational activities which are described as follows:

- Provisioning: This is the process of on-boarding users to systems and applications. These processes provide users with necessary access to data and technology resources. The term typically is used in reference to enterprise-level resource management. Provisioning can be thought of as a combination of the duties of the human resources and IT departments, where users are given access to data repositories or systems, applications, and databases based on a unique user identity. Deprovisioning works in the opposite manner, resulting in the deletion or deactivation of an identity or of privileges assigned to the user identity.
- Credential and attribute management: These processes are designed to manage the life cycle of credentials and user attributes—create, issue, manage, revoke—to minimize the business risk associated with identity impersonation and inappropriate account use. Credentials are usually bound to an individual and are verified during the authentication process. The processes include provisioning of attributes, static (e.g., standard text password) and dynamic (e.g., one-time password) credentials that comply with a password standard (e.g., passwords resistant to dictionary attacks), handling password expiration, encryption management of credentials during transit and at rest, and access policies of user attributes (privacy and handling of attributes for various regulatory reasons).
- Entitlement management: Entitlements are also referred to as authorization policies. The processes in this domain address the provisioning and deprovisioning of privileges needed for the user to access resources including systems, applications, and databases. Proper entitlement management ensures that users are assigned only the required privileges (least privileges) that match with their job functions. Entitlement management can be used to strengthen the security of web services, web applications, legacy applications, documents and files, and physical security systems.
- Compliance management: This process implies that access rights and privileges are monitored and tracked to ensure the security of an enterprise's resources. The process also helps auditors verify compliance to various internal access control policies, and standards that include practices such as segregation of duties, access monitoring, periodic auditing, and reporting. An example is a user certification process that allows application owners to certify that only authorized users have the privileges necessary to access business-sensitive information.
- Identity federation management: Federation is the process of managing the trust relationships established beyond the internal network boundaries or administrative domain boundaries among distinct organizations. A federation is an association of organizations that come together to exchange information about their users and resources to enable collaborations and transactions (e.g., sharing user information with the organizations' benefits systems managed by a third-party provider). Federation of identities to service providers will support SSO to cloud services.
- Centralization of authentication (authN) and authorization (authZ): A central authentication and authorization infrastructure alleviates the need for application developers to build custom authentication and authorization features into their applications. Furthermore, it promotes a loose coupling architecture where applications become agnostic to the authentication methods and policies. This approach is also called an "externalization of authN and authZ" from applications.

The following diagram shows the relationships between the basic functionalities and the operational activities during the lifecycle of the users within the system:

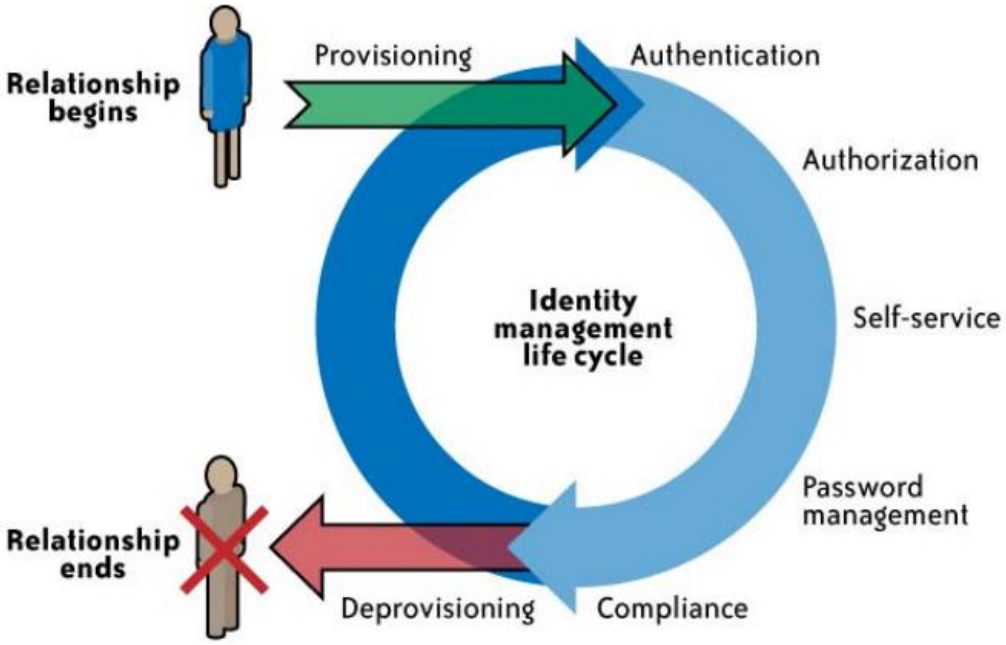

Figure 8: Identity and Access Management Life Cycle

3.8.2 Security Services

This chapter gathers the security services of the most popular cloud service providers. We will in the following consider in detail the following providers and security services:

- Amazon Security Services
  - AWS Identity and Access Management (IAM)
  - AWS Directory Service
  - Amazon Inspector (Preview)
  - AWS CloudHSM
  - AWS Key Management Service

- Microsoft Azure
  - Azure Active Directory
  - Multi-Factor Authentication
  - Key Vault
  - Virtual Network

- IBM SmartCloud
  - IBM Cloud Data Encryption Services: Secure
  - OnCloud Identity and Access Management
  - Vormetric Transparent Encryption
  - Identity Mixer

- Google Cloud Platform
  - Data Encryption
  - Secure Global Network
  - Intrusion Detection
  - Security Scanning
  - Compliance and Certifications

- Oracle Cloud

- Other providers
  - Ping Identity
  - AlertLogic
  - CipherCloud
  - CloudLock
  - Qualys

3.8.2.1 Amazon Security Services

3.8.2.1.1 AWS Identity and Access Management (IAM)

Link: https://aws.amazon.com/iam/

Description: *AWS Identity and Access Management (IAM) enables you to securely control access to AWS services and resources for your users. Using IAM, you can create and manage AWS users and groups, and use permissions to allow and deny their access to AWS resources.*

Conclusions: The main advantage of the AWS IAM is the complete integration with other Amazon Cloud services. A normal Identity Access Manager allows the creation and management of users, groups and roles and those permissions associated with them. However it is focused on the users, groups or roles in charge of developing, deploying and running the Cloud Services (managers, developers, client services, etc.) but it does not give solutions for the final end users.

3.8.2.1.2 AWS Directory Service

Link: https://aws.amazon.com/directoryservice/

Description: *AWS Directory Service is a managed service that allows you to connect your AWS resources with an existing on-premises Microsoft Active Directory or to set up a new, stand-alone directory in the AWS cloud. Connecting to an on-premises directory is easy and once this connection is established, all users can access AWS resources and applications with their existing corporate credentials. You can also launch managed, Samba-based directories in a matter of minutes, simplifying the deployment and management of Linux and Microsoft Windows workloads in the AWS cloud.*

Conclusion: This service allows having a LDAP database within the cloud and integrating it with the existing one on the premises of the client. The main advantage of this service is at the same time their worst facet because it is using Microsoft Active Directory instead of a universal LDAP interface, with all the advantages and disadvantages that Microsoft Active Directory brings with it.

3.8.2.1.3 Amazon Inspector (Preview)

Link: https://aws.amazon.com/inspector/

Description: *Amazon Inspector is an automated security assessment service that helps improve the security and compliance of applications deployed on AWS. Amazon Inspector automatically assesses applications for vulnerabilities or deviations from best practices. After performing an assessment, Amazon Inspector produces a detailed report with prioritized steps for remediation. To help you get started quickly, Amazon Inspector includes a knowledge base of hundreds of rules mapped to common security compliance standards (e.g. PCI DSS) and vulnerability definitions.*

Conclusions: Although this service it not related directly with the main factors of our investigation (Data Storage and Identity Access Manager), the Consortium has considered it important enough to be included in this study. This service can scan the services deployed within the cloud, based on the broad database of vulnerabilities and to find any security vulnerability on the services a company provides.

3.8.2.1.4 AWS CloudHSM

Link: https://aws.amazon.com/cloudhsm/

Description: *The AWS CloudHSM service helps you meet corporate, contractual and regulatory compliance requirements for data security by using dedicated Hardware Security Module (HSM) appliances within the AWS cloud. With CloudHSM, you control the encryption keys and cryptographic operations performed by the HSM.AWS and AWS Marketplace partners offer a variety of solutions for protecting sensitive data within the AWS platform, but for applications and data subject to rigorous contractual or regulatory requirements for managing cryptographic keys, additional protection is sometimes necessary. Until now, your only option was to store the sensitive data (or the encryption keys protecting the sensitive data) in your on-premises datacenters. Unfortunately, this either prevented you from migrating these applications to the cloud or significantly slowed their performance. The AWS CloudHSM service allows you to protect your encryption keys within HSMs designed and validated to government standards for secure key management. You can securely generate, store, and manage the cryptographic keys used for data encryption such that they are accessible only by you. AWS CloudHSM helps you comply with strict key management requirements without sacrificing application performance.*

The AWS CloudHSM service works with Amazon Virtual Private Cloud (VPC). CloudHSM instances are provisioned inside your VPC with an IP address that you specify, providing simple and private network connectivity to your Amazon Elastic Compute Cloud (EC2) instances. Placing CloudHSM instances near your EC2 instances decreases network latency, which can improve application performance. AWS provides dedicated and exclusive (single tenant) access to CloudHSM instances, isolated from other AWS customers. Available in multiple Regions and Availability Zones (AZs), AWS CloudHSM allows you to add secure and durable key storage to your applications.

Conclusions: This is a solution provided by Amazon for those systems dealing with sensitive data. This solution includes different security aspects: including your services within a VPN, encrypting the data using a PKI infrastructure and providing a separated infrastructure to access the PKI keys in order to improve the performance of the system. This solution simplifies the security controls however its biggest disadvantage is the price (AWS CloudHSM service with Dedicated SafeNet Luna SA is $5,000, plus $1.96 per Hour)

3.8.2.1.5 AWS Key Management Service

Link: https://aws.amazon.com/kms/

Description: *AWS Key Management Service (KMS) is a managed service that makes it easy for you to create and control the encryption keys used to encrypt your data, and uses Hardware Security Modules (HSMs) to protect the security of your keys. AWS Key Management Service is integrated with several other AWS services to help you protect your data you store with these services. AWS Key Management Service is also integrated with AWS CloudTrail to provide you with logs of all key usage to help meet your regulatory and compliance needs.*

Conclusions: This service was one of those included within the solution described previously. It allows storing the PKI keys within a centralized place that makes management of them easier. The biggest advantage that this solution claims is that it improves the performance of the system in corporation with other solutions that store the PKI infrastructure within a company's facilities.

3.8.2.2 Microsoft Azure

https://azure.microsoft.com/en-us/support/trust-center/security/

3.8.2.2.1 Azure Active Directory

Link: https://azure.microsoft.com/en-us/services/active-directory/

Description: *Azure Active Directory is a comprehensive identity and access management cloud solution that provides a robust set of capabilities to manage users and groups. It helps secure access to on-premises and cloud applications, including Microsoft online services like Office 365 and many non-Microsoft SaaS applications. Azure Active Directory comes in 3 editions: Free, Basic, and Premium.*

Conclusions: Azure Active Directory is a solution that covers the main services associated with an IAM (Authentication, Authorization and Auditing). In addition it covers the provisioning operational activities although the documentation is not clear about other operational activities, for instance de-provisioning is not even mentioned in the official documentation. One of the main disadvantages is that it is strongly related with the Microsoft products and, for instance it allows the set-up of end users if they are going to use Microsoft products, such as: Microsoft 365, Salesforce, Sharepoint, etc. but the integration with other products is not detailed in the documentation.

3.8.2.2.2 Multi-Factor Authentication

Link: https://azure.microsoft.com/en-us/services/multi-factor-authentication/

Description: *Azure Multi-Factor Authentication helps safeguard access to data and applications while meeting user demand for a simple sign-in process. It delivers strong authentication with a range of easy verification options—phone call, text message, or mobile app notification—allowing users to choose the method they prefer.*

Conclusions: This solution is an extension of the service described above and it provides some multi-factor authentication services. Its stronger advantage is that it can be configured with a simple click and provides a SDK to be integrated with your own applications. However, as described with the Active Directory Service, the interrelationship with Microsoft products is very strong and it is not clear that all types all developments will suit this solution.

3.8.2.2.3 Key Vault

Link: https://azure.microsoft.com/en-us/services/key-vault/

Description: *Secure key management is essential to protecting data in the cloud. With Azure Key Vault, you can encrypt keys and small secrets like passwords using keys stored in hardware security modules (HSMs). For added assurance, import or generate your keys in HSMs that are validated to FIPS 140-2 Level 2 standards—so that your keys stay within the HSM boundary. Key Vault is designed so that Microsoft does not see or extract your keys. Monitor and audit key use with Azure logging—pipe logs into Azure HDInsight or your SIEM for additional analysis and threat detection (coming soon).*

Conclusions: As Amazon, Microsoft provides a PKI store manager service that allows the clients to use this infrastructure directly in the cloud which improved performance of the system considerably.

3.8.2.2.4 Virtual Network

Link: https://azure.microsoft.com/en-us/services/virtual-network/

Description: *Azure Virtual Network provides an isolated and secure environment to run your virtual machines and applications. You can use your private IP addresses and define subnets, access control policies, and more. With Virtual Networks, you can treat Azure just as you would your own datacenter.*

Conclusions: Virtual Network provides either IPSec or VPN functionalities, allowing the insolation of those desired services or servers. This solution relies on the Authentication and Authorization services of Microsoft Active Directory solution (described before) therefore and once again the relationship between Microsoft products is very strong.

3.8.2.3 IBM SmartCloud

3.8.2.3.1 IBM Cloud Data Encryption Services: Secure

Link: https://marketplace.ibmcloud.com/apps/2461#!overview

Description: *IBM Cloud Data Encryption Services (ICDES) is a software defined data protection offering that runs in the background of your application servers. It's cryptographic splitting combines AES-256 certified encryption with randomized (keyed) information dispersal into an easy to use FIPS 140-2 certified solution. ICDES has a built-in simplified key management system, so no large, expensive key storage systems are required. ICDES can help support regulatory compliance requirements for HIPAA, HITECH, FISMA, Sarbanes-Oxley, and PCI.*

Conclusions: IBM provides this service for those specific applications with an encryption need. In contrast with similar services described above (Amazon or Microsoft), this solution includes the PKI management in the cloud integrated within the service with savings associated with not having to hire a new service. However due to American legal restrictions on the encryption, the encryption algorithm provided by this service (e.g. *AES-256)* seems a little bit weak.

3.8.2.3.2 OnCloud Identity and Access Management

Link: https://marketplace.ibmcloud.com/apps/5000?restoreSearch=true#!overview

Description: *OnCloud is an Identity and Access Management-as-a-Service (IAMaaS) solution available to Commercial and Government customers. OnCloud utilizes IBM's market-leading Identity and Access Management (IAM) technology to provide a full and adaptable set of IAM capabilities, including Identity Governance, Strong Authentication, Access Control, Single Sign-On, and Federation. OnCloud is infrastructure agnostic, and can be deployed on the customer's infrastructure of Untitled choice.*

Conclusions: This service provides the basic set of Authentication, Authorization and Auditing. Its biggest advantage is its price as it is free with unlimited users. In addition it includes the strong authentication functionalities that in other cloud providers are considered as another service therefore you need to pay for it separately. However it seems it is focused on the administrators and developers of the cloud. Furthermore the documentation available doesn't describe or detail how to link the functionalities available with end users to use your own applications.

3.8.2.3.3 Vormetric Transparent Encryption

Link: https://marketplace.ibmcloud.com/apps/292?restoreSearch=true#!overview

Description: *Vormetric addresses industry compliance mandates and government regulations globally by securing data in physical, virtual and cloud infrastructures, through: Data Encryption, Key Management, Access Policies, Privileged User Control, and Security Intelligence.*

Conclusions: Vormetric Transparent Encryption allows not only the encryption of the data storage in the cloud but also the establishment of authorization policies to access this encrypted data. Unfortunately the documentation available for this service is vague and it makes it very difficult to evaluate the functionalities that this service provides.

3.8.2.3.4 Identity Mixer

Link: http://www.zurich.ibm.com/idemix/

Description: *We all use electronic services increasingly often in our daily lives. To do so, we have no choice but to provide plenty of personal information for authorization, billing purposes, or as part of the terms and conditions of service providers.*

Dispersing all this personal information erodes our privacy and puts us at risk of abuse of this information by criminals.

Identity Mixer allows users to minimize the personal data they have to reveal in such transactions. For instance, if electronic identity (eID) cards were realized with Identity Mixer, then teenagers possessing such eID cards could log onto a teenage chat room just proving that they are indeed 12–15 years of age without revealing any other information stored on the card such as their name or address.

Conclusions: Identity Mixer should have a special mention as it is a service provided by IBM that was born as an FP7 project. This solution is not the usual service providing encryption capabilities and it is going a step beyond that. The main aim of this service is to provide a specific security control for the privacy of personal data. Therefore this service could be a very good reference for the purposes of this PRISMACLOUD project.

3.8.2.4 Google Cloud Platform

Unfortunately the website provided by Google Cloud Platform doesn't describe in detail the security services that they provide. The only Security information detailed in the website is an abstract description of the different functionalities. This information is available using the following link:

https://cloud.google.com/security/#platform_security_features

As a brief summary of the security information available, the most relevant descriptions for the purposes of PRISMACLOUD are as follows:

Data Encryption

Cloud Platform services automatically encrypt data before it is written to disk. For example, the data for each Cloud Storage object and its metadata is encrypted under the 256-bit Advanced Encryption Standard, and each encryption key is itself encrypted with a regularly rotated set of master keys. The same encryption and key management policies used for your data in Cloud Platform are used by many of Google's production services, including Google Docs, Gmail, and Google's own corporate data.

Secure Global Network

Because it's linked to most ISPs in the world, Google's global network helps to improve the security of data in transit by limiting hops across the public Internet. Cloud Interconnect and managed VPN allow you to create encrypted channels between your private IP environment on premises and Google's network. This allows you to keep instances completely disconnected from the public internet while still reachable from your own private infrastructure.

Intrusion Detection

Google intrusion detection involves tightly controlling the size and make-up of Google's attack surface through preventative measures, employing intelligent detection controls at data entry points, and employing technologies that automatically remedy certain dangerous situations.

Security Scanning

Cloud Security Scanner helps App Engine developers identify the most common vulnerabilities, specifically cross-site scripting (XSS) and mixed content, in their web applications.

Compliance and Certifications

Cloud Platform and Google infrastructure is certified for a growing number of compliance standards and controls. Read more about the specific certifications on our compliance page.

3.8.2.5 Oracle Cloud

If the security documentation provided by Google can be qualified as poor, then the Security documentation provided by Oracle Cloud is almost non-existent and it is barely mentioned on its website. In addition the Oracle Cloud website doesn't provide any services related to security or indeed what security processes they are following.

3.8.2.6 Other providers

This study has also taken into consideration other security services that can be used within the cloud and they are not associated with the main Cloud providers. These security services have been selected using, as a starting point, the classification of services defined in Cloud Services Taxonomy provided by OpenCrowd [31]. This Taxonomy classifies the different Cloud Services into functional blocks and the Software Services related with Security. They are as follows:

3.8.2.6.1 Ping Identity

Link: https://www.pingidentity.com/products/

Description: *PingID™ is a multi-factor authentication solution that enables secure sign-on to all of your applications. Working with PingOne® and PingFederate®, as well as VPN servers, PingID:*

- Defines and enforces authentication policies tailored to your needs.
- Is delivered via an easy-to-use mobile application.
- Secures your applications without passwords.

Conclusions: PingID is an Identity Access Manager with Single Sign-On functionalities. The biggest advantage provided by PingID is that it allows the use of strong authentication using contextual information that makes it almost transparent to the end user that he/she is using strong authentication. In addition it is able to provide IAM services to all types of users, including the end users of the services. Nevertheless the integration of your own applications within the system is defined using an undocumented API therefore it cannot be evaluated by this study.

3.8.2.6.2 AlertLogic

Link: https://www.alertlogic.com/solutions/alertlogic-technology/activewatch/

Description: *Effective cloud security begins with data collection, but your data is only as good as the insight it provides. Alert Logic ActiveWatch is a managed Security-as-a-Service (SaaS) solution providing the human expertise required for deep insight into your security and compliance posture.*

3.8.2.6.3 CipherCloud

Link: http://www.ciphercloud.com/

Description: *The CipherCloud Open Platform eliminates cloud security issues by delivering a single platform to secure sensitive customer information across all of your cloud applications, while preserving usability, functionality and performance. Available as a service or virtual appliance, CipherCloud delivers a comprehensive set of protection controls including encryption, tokenization, activity monitoring, data loss prevention (DLP) and malware detection that can overcome your cloud security concerns.*

Conclusions: CipherCloud provides a set of security services where it is worth mentioning that those services ensure data privacy on the data residency and they claim that they adhere to regulatory

[31] http://cloudtaxonomy.opencrowd.com/taxonomy/about/

compliance. The documentation available in their website does not make it very clear how CipherCloud achieves this.

3.8.2.6.4 CloudLock

Link: http://www.cloudlock.com/

Description: CloudLock provides compliance and security solutions for enterprises using public cloud applications like Salesforce and Google Apps. The largest organizations in the world trust CloudLock to secure their data.

Conclusions: CloudLock offers a set of different products focused on the discovering and monitoring of possible cybersecurity threats in order to reduce or minimize the security risk, establishing safeguards around apps that access accounts and data. Therefore the services provided by CloudLock are out of the scope of this study.

3.8.2.6.5 Qualys

Link: http://www.qualys.com/

Description: *Qualys, Inc. is the leading provider of on demand IT security risk and compliance management solutions — delivered as a service. Qualys' Software-as-a-Service solutions are deployed in a matter of hours anywhere in the world, providing customers an immediate and continuous view of their security and compliance postures. The QualysGuard® service is used today by more than 5,000 organizations in 85 countries, including 50 of the Fortune 100 and performs over 500 million IP audits per year. Qualys has the largest vulnerability management deployment in the world at a leading global company.*

Conclusions: As the services described previously, the products provided by Qualys Inc. are focused on the detection or identification, classification and control of possible cybersecurity threats. Hence these services are out of the scope of this study.

4 Major benefits and risks in cloud computing

This chapter will provide high-level details related to the Cloud Computing landscape with focus on the aspects related to security. The section is composed of two sub-sections, the first exploring the potentialities (i.e. the added value) of the security framework provided by Cloud computing and part of its cloud service offerings with respect to traditional on-premises security solutions, while the second sub-section will explore the main concerns and the main risk/threats that Customer must take into account when entrusting a cloud computing solution.

4.1 Cloud security benefits

Across all PRISMACLOUD investigation activities we have concentrated our analysis related to cloud security framework on the gaps of the actual cloud security infrastructure, but it is widely recognized that Cloud security must not be perceived only as a concern but also as a potentiality. In fact Cloud-based security offers a certain number benefits over the traditional premise-based model which can be summarized in the following main key topics:

- **less capital expenditure** (security implemented at a large scale is cheaper, Cloud Provider leveraging on large scale deployment can re-invest the CAPEX savings to refresh the security equipment, thus been able to deploy state-of-art of security top technologies)
- **enhanced efficiency** (The centralization of security policy in the set of devices which constitutes the defence perimeter of Data Center allows for a more manageable and secure infrastructure framework)
- **Higher level of security** (as it's proactively managed by a team of security experts).
- **Security patches campaign** (The centralized nature of the cloud security framework allows for an easier security patch management, thus resulting in more solid and more updated security framework)
- **Higher level of security with faster response to security incidents** (it is a common practice for Cloud Providers organization to have a dedicated Security NOC composed of high-skilled people who can quickly take care of security incidents.)
- **Security as a market differentiator** (security is a priority concern for many cloud customers; many of them will make buying choices on the basis of the reputation for confidentiality, integrity and resilience of, and the security services offered by, a provider. This is a strong driver for cloud providers to improve security practices.)
- **Rapid, smart scaling of resources** (the ability of the cloud provider to dynamically reallocate resources for filtering, traffic shaping, authentication, encryption, etc, to defensive measures (e.g., against DDoS attacks) has obvious advantages for resilience.)
- **Benefits of resource concentration** (Although the concentration of resources undoubtedly has disadvantages for security [see Risks], it has the obvious advantage of cheaper physical perimiterisation and physical access control (per unit resource) and the easier and cheaper application of many security-related processes.)

4.2 Cloud computing risks and threats

The most important cloud-specific risks are can be divided in three main classes, being **policy/organizational risks**, **technical risks**, **data protection risks**.

Over the years many attempts to provide an organized rationale related to cloud security policy and organisational risks, as well as cloud technical risks have been carried out—the most authoritative of them being the Cloud Computing Security Risk Assessment proposed by ENISA [16]. The assessment contains a regularly updated list of the top security risks related to Cloud computing.

4.2.1 Policy and organisational risks

- **Loss of governance:** Cloud computing framework forecast the situation where the Customer is willing to offload its own IT infrastructure to Cloud Provider thus inherently entrusting the control to the Cloud Provider. This situation can introduce a certain number of risks for Customer related to security matters:
    - Terms of use (of the cloud service) may prohibit e.g. pen testing
    - Conflict between customer hardening procedures and cloud environment
    - Terms and conditions may change due to CP outsourcing or CP change of control
    - Challenge compliance with certifications (e.g. PCI DSS payment card industry data security standard)
    - cloud provider (CP) cannot provide evidence on compliance with relevant requirements
    - CP does not provide audit by cloud customer

  The loss of governance and control could have a potentially severe impact on the organization's strategy and therefore on the capacity to meet its mission and goals. The loss of control and governance can also lead to the impossibility of complying with the security requirements, a lack of confidentiality, integrity and availability of data, and a deterioration of performance and quality of service, not to mention the introduction of compliance challenges, as certain organisations migrating to the cloud have made considerable investments in achieving certification either for competitive

- **Lock-in:** there still is little on offer in the way of tools, procedures or standard data formats or services interfaces that could guarantee data, application and service portability. This can make it difficult for the customer to migrate from one provider to another or migrate data and services back to an in-house IT environment. This introduces a dependency on a particular CP for service provision, especially if data portability, as the most fundamental aspect, is not enabled.
- Lack of interoperability (cf. also [17]) -> standardisation
- CPs may have an incentive to prevent portability
- Insolvency of CP may lead to "catastrophic business failure"
- Acquisition of CP (change in policy)
- PaaS lock-in at the API layer (e.g. incompatible APIs for data access)
- Paas lock-in at runtime layer (e.g. Modified Java runtime)

- IaaS lock-in: incompatible formats for packaging and distributing applications
- Bank-run-scenario if CP threatens to go out of service (some customers may not be able to extract their data and applications)

4.2.2 Technical risks

- **Isolation failure:** multi-tenancy and shared resources are defining characteristics of cloud computing. This risk category covers the failure of mechanisms separating storage, memory, routing and reputation between different tenants (e.g., so-called guest-hopping attacks). However it should be considered that attacks on resource isolation mechanisms (e.g. against hypervisors) are still less numerous and much more difficult for an attacker to put in practice compared to attacks on traditional OSs.
- **Management interface compromise:** customer management interfaces of a public cloud provider are accessible through the Internet and mediate access to larger sets of resources (than traditional hosting providers) and therefore pose an increased risk, especially when combined with remote access and web browser vulnerabilities.
- **Data protection:** cloud computing poses several data protection risks for cloud customers and providers. In some cases, it may be difficult for the cloud customer (in its role as data controller) to effectively check the data handling practices of the cloud provider and thus to be sure that the data is handled in a lawful way. This problem is exacerbated in cases of multiple transfers of data, e.g., between federated clouds. On the other hand, some cloud providers do provide information on their data handling practices. Some also offer certification summaries on their data processing and data security activities and the data controls they have in place, e.g., SAS70 certification.
- **Insecure or incomplete data deletion:** when a request to delete a cloud resource is made, as with most operating systems, this may not result in true wiping of the data. Adequate or timely data deletion may also be impossible (or undesirable from a customer perspective), either because extra copies of data are stored but are not available, or because the disk to be destroyed also stores data from other clients. In the case of multiple tenancies and the reuse of hardware resources, this represents a higher risk to the customer than with dedicated hardware.
- **Malicious insider:** while usually less likely, the damage which may be caused by malicious insiders is often far greater. Cloud architectures necessitate certain roles which are extremely high-risk. Examples include CP system administrators and managed security service providers.
- **Customers' security expectations:** the perception of Security levels by Customers might differentiate from the actual security (and availability) offered by the CP, or the actual temptation of the CP to reduce costs further by sacrificing on some security aspects.
- **Availability Chain:** Reliance on Internet Connectivity at Customer's end creates a Single point of failure in many cases.

On the other side, Cloud Security Alliance [18]conducted a survey of industry experts to compile professional opinion on the greatest vulnerabilities within cloud computing to identify the top threats. The **Top Threats** working group used these survey results alongside their expertise to craft the final

2013 report. The survey methodology validated that the threat listing reflects the most current concerns of the industry. In this most recent edition of this report, experts identified the following nine critical threats to cloud security (ranked in order of severity):

- **Data Breaches,** which is the intentional or unintentional release of secure information to an untrusted environment)
- **Data Loss**, which is an error condition in information systems in which information is destroyed by failures or neglect in storage, transmission, or processing
- **Account hijacking**, which occurs when an individual or organization's cloud account is stolen or hijacked by an attacker. Cloud account hijacking is a common tactic in identity theft schemes. The attacker uses the stolen account information to conduct malicious or unauthorized activity
- **Insecure APIs**, vulnerabilities in application programming interfaces to gain access to enterprise websites and networks and carry out other malicious activities
- **Denial of Service**, which is an attempt to make a machine or network resource unavailable to its intended users
- **Malicious Insiders**, which occurs when the employee, contractor or sub-contractor with access to data, files and IT systems who may be disgruntled or feel "obligated" to steal valuable intellectual property.
- **Abuse of Cloud Services**, which occurs when abusing the relative anonymity behind these registration and usage models, spammers, malicious code authors, and other criminals have been able to conduct their activities with relative impunity.
- **Insufficient Due Diligence**, which occurs when the Cloud provider is not able to provide the agreed level of Security controls information
- **Shared Technology Issues,** which arises when the Cloud provider is not able to guarantee proper security level for its customer relying over a multi-tenant shared infrastructure

4.2.3 Data protection risks

The Article 29 Working Party (launched according to Article 29 of the European Data Protection Directive 95/46/EC [19]) identifies in its Opinion 05/2012 on Cloud Computing [17] (p. 5ff) several data protection risks pertaining to personal data processing operations deploying cloud services:

- Lack of control
- Lack of availability due to lack of interoperability (vendor lock-in)
- Lack of integrity caused by the sharing of resources (conflicting interests/objectives of different customers)
- Lack of confidentiality in terms of law enforcement requests made directly to a cloud provider
- Lack of intervenability due to the complexity and dynamics of the outsourcing chain
- Lack of intervenability (data subjects' rights)
- Lack of isolation (administrators may link information of different clients)
- Lack on information on processing (transparency)

- Chain processing is taking place involving multiple processors and subcontractors
- Personal data are processed in different geographic locations within the European Economic Area (EEA)
- Personal data is transferred to third countries outside the EEA

5 PRISMACLOUD cloud security patterns

5.1 Introduction

5.1.1 Representation of knowledge in design patterns

The Vienna, Austria born Christopher Alexander, who has since 1963 been living and teaching in Berkeley, California[32], published his book "A Pattern Language: Towns, Buildings, Construction" [20] in 1977, where he (and his co-authors) introduced the concept of re-usable design solutions for architectural problems[33]. The idea behind the **architectural patterns** is to provide a collection of proven solutions for problems which occur over and over again. The 253 presented patterns contain the concentrated knowledge and experience of designers and are intended to be re-used. Alexander defines a **pattern language** as a collection of patterns from a specific domain. He and his co-authors used to order the **design patterns** they initially presented as result of the knowledge they collected, "beginning with the very largest, for regions and towns, then working down through neighbourhoods, clusters of buildings, buildings, rooms and alcoves, ending finally with detail of construction" [20]. This way, "smaller" patterns shall help complete "larger" patterns. The patterns proposed by Alexander were intended to be "alive and evolving". He viewed them as "hypotheses", as "current best guess", to be improved and possibly replaced with more profound patterns, as a result of "new experience and observation".

The idea of design patterns was taken up again in 1994 by computer scientists and especially software engineers who tried to tackle the reusability of software with a **software design pattern** approach. Reusability of software was then, after about 20 years of object oriented design a big issue. The resulting book" Design Patterns: Elements of Reusable Object-Oriented Software" [21] has become a standard and has not lost its significance and relevance in software engineering today. The problem setting in software engineering is comparable to that in the field of architecture: Not to "solve every problem from first principles" but instead use a proven solution to a design problem.

The idea of design patterns was applied to other contexts as well. **Security patterns**, or security design patterns "codify basic security knowledge in a structured and understandable way" [22]. They represent a practical means to communicate end user needs and requirements. Security patterns are connected to one or more specific security goals. A comprehensive collection of security patterns which were discussed at the annual "Pattern Languages of Programs" (PLoP) conferences since 1997, is available on the homepage of the security researcher Munawar Hafiz (Auburn University, Alabama, USA)[34]. It currently contains a catalogue of 97 security patterns.

[32] https://en.wikipedia.org/wiki/Christopher_Alexander

[33] It shall here be noted, that the book is a very pleasurable and most interesting read, particularly for anyone who lives in a house or apartment. The book also provides valuable insight for those who use the concept in design and security patterns. (The entire book – 1218 pages – can be found as pdf on the internet https://archive.org/details/APatternLanguage)

[34] http://www.munawarhafiz.com/securitypatterncatalog/index.php. Munawar Hafiz is also author of several papers on security patterns, e.g. [51], which presents "4 design patterns that can aide (sic!) the decision making process for the designers of privacy protecting systems".

There is also on-going work on **privacy patterns**, which connect problems to solutions within the context of user privacy. The ability of design patterns to communicate and address the often conflicting requirements from different actors in different domains, is ideal for their application in designing information privacy into information systems: "Privacy Patterns that span across usability, engineering, security and other considerations can provide sharable descriptions of generative solutions to common design contentions. Since patterns focus on describing the resolutions of contradictory forces in a design context, the pros and cons of a specific solution can be easily debated. Unlike guidelines, regulations or best practices, patterns are descriptive, rather than normative, facilitating discussion and debate and providing education rather than insisting on particular solutions or practices" [23][35]. There are several websites online for joint development of privacy design patterns, like privacypatterns.org[36] by researchers of the University of California, Berkeley, School of Law (funded with grants from the U.S. Department of Homeland Security and from the NIST, among others), and the privacypatterns.eu[37]—resulting from the European FP7 project PRIPARE[38] (Preparing industry to privacy-by-design by supporting its application in research).

The concept of design patterns has also been specifically applied to cloud computing, and several collections of cloud computing patterns are available online on the internet[39] but without a particular focus on information security.

5.1.2 PRISMACLOUD crypto primitives

The PRISMACLOUD project proposes a set of several **cryptographic primitives** (crypto-tools) for enabling end to end security in cloud applications and services, countering some of the most pressing threats currently present in cloud computing. Cryptographic primitives are basic cryptographic functions (or algorithms) which can be used in cryptographic protocols in security relevant information and communication technology (ICT) applications. A **cryptographic protocol** can be defined as an exact description of how a specific cryptographic functionality is carried out: it describes the exact steps of application of the cryptographic algorithms, as well as the structure of the data on which the algorithms operate. Without exception, the PRISMACLOUD cryptographic primitives are either extensions of existing cryptographic primitives (where they add functionality and/or cryptographic strength), or security enhancements of functions that were not equipped with security functionalities before.

The PRISMACLOUD cryptographic primitives are from **four specific fields** in which security and privacy issues are pending in current cloud applications and services. These are the fields of **data storage in the cloud**, of **user privacy protection and data minimisation,** of **authentication of stored and processed data**, and of **certification of virtualised infrastructures**.

[35] In this article the authors apply the concept of design patterns to express so-called "privacy design anti-patterns" for misapplied patterns, with unintended consequences.
[36] http://privacypatterns.org/
[37] https://privacypatterns.eu/
[38] http://pripareproject.eu/
[39] Two websites which are published in parallel to books, present extensive catalogues of cloud (security) patterns: http://www.cloudcomputingpatterns.org/ belongs to [52], while http://cloudpatterns.org/ tries to collect patterns for [53] (to be published 2016).

The following list presents the single cryptographic primitives by number and name, with which they will be referenced in the "solution" category of the single patterns:

Data storage in the cloud:

- Primitive 1: Cryptographic storage solution
- Primitive 2: Data security for database applications

User privacy protection and data minimisation:

- Primitive 3: Anonymous credentials
- Primitive 4: Group signatures
- Primitive 5: Big data anonymisation

Authentication of stored and processed data:

- Primitive 6: Malleable signatures
- Primitive 7: Verifiable computation

Certification of virtualised infrastructures:

- Primitive 8: Certification of virtualised infrastructures

Table 4: PRISMACLOUD cryptographic primitives

5.1.3 Assumptions for the proposed cloud security patterns

In the following, we will present assumptions and prerequisites we intend to use for our proposed cloud security patterns:

- The cloud security patterns do not represent "hard requirements" on cloud applications and services, the patterns represent more a way of communicating a user need (and specifically a security need) to the system architects and developers of the services in an informal way. The system architects and developers shall read from the pattern the information that enables them to develop the cryptographic primitives in such a way, that the applications and systems where they are used, satisfy the security needs of the end-user.
- In our particular case, the definition and selection of the cryptographic primitives belonging to the single cloud security patterns was not carried out *after* the development of the security patterns, where a specific need for these primitives would have been expressed. The cryptographic primitives, for which we now want to develop the corresponding cloud security patterns, were selected by the authors of the project proposal from among most recent cryptographic security techniques to counter some of the most dangerous threats in the following cloud computing areas:
    - Data storage in the cloud (securing data at rest),
    - Authentication of stored and processed data,
    - Certification of virtualised infrastructures, and
    - User privacy protection.

The cryptographic primitives were selected by the authors of the project proposal with specific security functionalities in mind. On these functionalities we base corresponding cloud security patterns, which we want to develop in order to:

- Give an alternative definition of PRISMACLOUD's crypto primitives in a way which is compatible and accessible both by the cryptographers and system engineers, who are developing the crypto primitives from the basic cryptographic functions, and the prospective end users.
- Potentially discover additional requirements from the end-user side,
- Provide a commonly understandable description of complex cryptographic functions to communicate the project innovations to a wider audience (both in related disciplines, as e.g. standardisation of cryptographic technologies, and in the realm of end-users, policy makers, and interested individuals)

5.1.4 Pattern categories description

Different publications about security patterns (and about design patterns in general) define the patterns along different categories. We have taken into consideration the categories used in [20], [21], [22], as well the categories used on the security pattern websites *securitypatterncatalog*, *cloudcomputingpatterns*(links in the footnotes on the previous pages) and have chosen a synthesis that seems suitable for us. We use the same main categories as in Alexander's (et al.) seminal pattern book [20] (problem, solution, etc.), as do all the other sources. Alexander et al. are very concise and enlightening in explaining the single categories. So their descriptions—although they were used to describe an architectural pattern language—are analogously used for the description of our categories, where it is applicable in the context of cloud security[40]. Several other categories, most of them taken from the other sources and the websites, are also mentioned in [square brackets]. They will be used in the description of the single patterns, where this kind of information is valuable for the description of circumstances or for the communication of user needs.

We use the following categories and instructions to characterise the patterns:

- **Name**: Name and
- **Summary description** of the pattern;
- **[Also known as**: Other names]
- **[Example**: Present a real world example, illustrating the existence of the problem]
- **Context**: "Introductory paragraph"; describe, when the pattern is applicable; "how it helps to complete certain larger patterns"
- **Intention**: list end user values, security properties covered for end user

[40] The citations without reference in the pattern categories are without exception from Alexander's et. al. book [20], p. X ff.

- **Problem**: first a (bold) headline with the "essence of the problem in one or two sentences"; then the "body of the problem": "empirical background of the pattern", "evidence for its validity", range of different manifestations

- **Solution**:"[T]he heart of the pattern—which describes the field of physical and social relationships which are required to solve the stated problem, in the stated context"; "The solution describes the elements that make up the design, their relationships, responsibilities, and collaborations"; "[T]he pattern provides an abstract description of a design problem and how a general arrangement of elements (…) solves it"[41]: Here we will link to the:

    - ☞ **PRISMACLOUD cryptographic primitive(s) upon which the solution is based** (solution principle), "this solution is always stated in the form of an instruction—so that you exactly know what you need to do, to build the pattern". describe the relationship which is required to solve the stated problem

- **[Diagram**: Alexander proposes to use a diagram at this point of the pattern; some of the security pattern sources use UML style diagrams for depicting information and interaction flow]

- **[Known uses**: Known uses of the pattern in existing systems}

- **Consequences**: list potential benefits and liabilities;

    - ☞ here we reference the "**threats which are countered"** by using that single patterns

The following table lists the defined cloud security patterns, together with the cryptographic primitive(s) upon which the solution is based:

[41] Both citations from [21] where design patterns are used in the context of re-usable object oriented software design.

Field 1: Data storage in the cloud

- Pattern 1: **Secure cloud storage by default**

  Primitive 1: Cryptographic storage solution

- Pattern 2: **Moving a legacy application's database to the cloud**

  Primitive 2: Data security for database applications

Field 2: User privacy protection and data minimisation

- Pattern 3: **Non-identifiable and untrackable use of a cloud service**

  Primitive 3: Anonymous credentials

- Pattern 4: **Minimise exposure of private data during authentication in the cloud**

  Primitive 3: Anonymous credentials

- Pattern 5: **Big data anonymisation**

  Primitive 5: Big data anonymisation

Field 3: Authentication of stored and processed data

- Pattern 6: **Protect the authenticity of a data set and possible subsets**

  Primitive 6: Malleable signatures

- Pattern 7: **Authorise controlled subsequent modifications of signed data**

  Primitive 6: Malleable signatures

- Pattern 8: **Controlling the correctness of delegated computations**

  Primitive: 7 Verifiable computation

Field 4: Certification of virtualised infrastructures

- Pattern 9: **Controlling your virtual infrastructures**

  Primitive 8: Certification of virtualized infrastructures

Table 5: PRISMACLOUD patterns and employed crypto primitives

5.2 Field 1: Data storage in the cloud

5.2.1 Pattern 1: Secure cloud storage by default

Name: Secure cloud storage by default

Summary description: This pattern describes the qualities of a cloud storage service, as most users would expect when moving their digital assets to the cloud: The data in the cloud storage remains readily available when needed, and dependably and securely confidential against the cloud provider and other tenants in the vicinity of the cloud, as well as against other third parties which are not entitled by the user to access the data. The data may easily be shared with others, and easily be transferred to another cloud provider when the user wants to do so.

[Also known as: "The dream of every cloud user"**]**

[Example:]

Context: The pattern is applicable in any context where a user wants to securely store or share a data object in a cloud infrastructure.

Intention: Provide a cloud storage service with strong[42] confidentiality, integrity, and availability, from which the cloud user can anytime effectively pull away the stored data.

Problem: Currently, most cloud storage providers store the data either unencrypted, or apply encryption which remains completely under their control; some cloud users locally encrypt their data before they store it in the cloud in order to maintain the **confidentiality** of the data.

Whether the cloud provider encrypts or does not encrypt the data it stores, the cloud provider has in practice full access to the data—if it is not encrypted by the user in the first place. In many cases, especially in free-of-charge public cloud services from the big cloud providers (in the fields of cloud file storage, combined with collaboration services, of cloud-based email and other communication services) the end users have to consent to terms-of-reference granting the providers full rights to the data (including rights to store, combine, or otherwise use the data in ways non-anticipated and not explicitly consented to by the user, to sell or commercialise the data in any other imaginable way). The users of such cloud services are in internet idiom also referred to as "net fish", which have to be nourished with gimmicks and entertainment that they continue to give away their data.

Nevertheless, also in "serious" cloud services (for customers that pay), the cloud provider has to be trusted to maintain the confidentiality of the data—by not looking at the data itself, and by effectively protecting the data in storage against access by unauthorised third parties. This includes all copies and replications of the data which are created for availability purposes in all layers of a storage architecture.

[42] "strong" refers here to cryptographic strength beyond computational assumptions, or otherwise bounded adversaries. The security of such cryptographic schemes is also called information theoretic security, or unconditional security; such systems remain secure even against unlimited computing resources for long periods of time.

Also with respect to **availability** of data and of cloud services, the user is dependent on the provider. There are cases known, where bankruptcy of a cloud provider led to sudden loss of access to customer data.

Deletion of data in clouds is also a big issue and it is not sufficiently solved how an effective deletion of data in all replications and backups can be achieved and substantiated.

When cloud users use end-to-end encryption to mitigate some of the mentioned problems and threats (i.e. when they encrypt the data before they store it in the cloud) they are required to implement and maintain a cryptographic key management system and an access control mechanism, with all its known complexities and implications.

Solution: If cloud users use a cloud storage, they do not want to give up their property rights and privacy rights on the data. Cloud users want to maintain **full control** over their cloud storage by default. They want **strong confidentiality guarantees by default** while being able to **share certain data** with other cloud users or with the cloud provider at their own discretion. The data needs to be **protected against loss** by some kind of redundancy in a way that the confidentiality remains upheld. The cloud user wants to be able to **withdraw the data from one cloud provider**, and give it to another provider for hosting, at any time without having to rely on any form of cooperation with the cloud provider. The cloud wants to be sure that the data can be completely withdrawn, with no copies of the information plain remaining at the provider.

☞ **PRISMACLOUD primitive:** PRISMACLOUD proposes a cryptographic storage solution with increased practical usability for the secure, distributed storage of data [24] [25]. Through the use of an information-dispersal algorithm, i.e. secret sharing [26], the information is split into a number of shares, of which any subset of a fixed number smaller than the number of shares allows the reconstruction of the original data. The numbers have to be selected at the time of storing the data and typically remain fixed throughout its lifetime. An example would be a threshold of 3 out of 5 system, where the data can be reconstructed using any 3 shares of the produced five shares. The 5 shares are distributed over encrypted channels to different cloud providers.

The cryptographic storage solution provides sort of a ‘keyless’ cryptographic solution, under the assumption, that not a number of cloud storage providers greater or equal the threshold of the storage network do maliciously cooperate (non-collusion assumption). The secret sharing algorithm itself is considerably stronger than commonly used cryptographic systems and capable of long-term security [27] and therefore applicable in scenarios with highest confidentiality requirements, like in e-Health or e-Government.

The cryptographic storage solution enables the collaboration of several users on the data, but it requires an explicit access control system;

The secret sharing also solves the availability problem at the user level, without the need of explicit backups. Also single shares can be taken out of the system and be replaced by newly generated ones. This prevents vendor lock-in and, when shares are continuously renewed, enables long-term data security as it minimises the chance of an attacker to get a sufficient number of shares for reconstructing the information by attacking one cloud provider after the other.

Leakage of metadata, which occurs during storage and retrieval of the single shares, and by synchronisation activity between the single storage providers during share renewal, may present a privacy problem and needs to be investigated.

[Diagram:]

[Known uses:]

Consequences: "Secure cloud storage by default" counters almost all identified risks related to confidentiality, integrity, and availability of stored data in the cloud and therefore constitutes a **disruptive technology** of highest potential. The application of that cryptographic primitive can potentially entirely transform cloud provisioning world-wide.

One new assumption which is introduced by this cryptographic primitive, is the **non-collusion assumption**, i.e. that sufficiently many of the cloud providers maliciously cooperate to discover the secret. This means, that the number of shares necessary to reconstruct the secret in the threshold scheme of the information dispersal algorithm is a crucial design parameter. The non-collusion assumption can only be substantiated by other assumptions on the trustworthiness of the single involved cloud providers. On the other hand, the data owner does not have to rely on computational assumptions for the confidentiality of the data. Detailed reference to the single risks will be given below.

☞ **Countered threats:**

The cloud security pattern covers all threats mentioned in chapter **Fehler! Verweisquelle konnte nicht gefunden werden.** "**Fehler! Verweisquelle konnte nicht gefunden werden.**" in the categories

Policy and organisational risks:

Loss of governance with respect to having the authority to effectively decide about access to the data, about moving the data, deleting the data.

Lock-in is effectively countered by the ability to exclude shares from the data set and to generate new shares to be stored at a different provider.

Technical risks:

Many of the technical risks are covered by the implicit encryption provided by the scheme, like **isolation failure, management interface compromise, data protection, Insecure or incomplete data deletion, malicious insider, customer's security expectation** – but also only under a non-collusion assumption: It has to be assumed that the single risks do not challenge a number of providers, above the threshold of the scheme. All the mentioned technical risks can be mitigated by continuous renewal and replacement of the shares. This reduces the window for an attacker for procuring a sufficient number of shares for reconstructing the unencrypted information. **Availability chain**, i.e. the reliance of the user on internet connectivity, is a risk which actually **improves** with a secret sharing primitive: Some storage provider may be off-line and the secret still be reconstructed with the shares from the remaining providers. On the other hand, if many providers would be off-line simultaneously, there might not be sufficient cloud providers to facilitate the reconstruction of the information. On the user side, like in all cloud systems, the user always has to rely on his/her internet connectivity.

Pattern 2: Moving a legacy application's database to the cloud

Name: Moving a legacy database application to the cloud

Summary description: This pattern enables the deployment of the database of a legacy application to the cloud while protecting the confidentiality of data that was not protected, when the database was hosted inside the security perimeter of the end user.

[Also known as:]

[Example:]

Context: The pattern is applicable when an end user wants to deploy an existing database to a public cloud.

Intention: Maintain confidentiality of data in legacy database applications, where the database itself is deployed to a public cloud.

Problem: Many businesses and administrations rely on legacy database applications which store data in a local, plain, unencrypted database—this is perfectly reasonable for a system deployed entirely within an organisation's security perimeter (i.e. in its own private datacenter). Now a business or administration for some reason wants to move the database to a public cloud. Out of compatibility and interoperability issues, or because they have a valid certification of the application in compliance with some regulation (which would require expensive re-certification or entirely new certification in case of modifications or re-implementation) the current application shall be used with the database which is now deployed to the cloud.

One problem is, that the data can no longer be stored in the database in plain text, as the database is deployed to the only partially trusted, potentially insecure infrastructure of the cloud provider and no longer hosted in the own trusted datacenter of e.g. the municipality or company that owns the application. Encrypting the entire database would usually require a different database design to adapt the fields for accepting cryptograms instead of clear values, and consequently also substantial modification of the application to remain functional.

Solution: Transparently add a layer of cryptography directly into the data fields of the database applications in such a way, that the encrypted data can be stored in the same fields as before.

☞ **PRISMACLOUD primitive:** PRISMACLOUD proposes to use the four cryptographic technologies:

- Format preserving encryption (FPE)
- Format preserving tokenisation (FPT)
- Order preserving encryption (OPE)
- Order preserving tokenisation (OPT)

FPE and FPT apply encryption in a manner such that the ciphertext has the same format as the plaintext (e.g. a social security number is mapped into a cryptogram with the format of a social security number) [28]. FPE uses an encryption algorithm to map clear data to encrypted data and there exist several algorithms for specific field types, like social security numbers etc. FPT uses tokenisation, which is using a lookup table to translate between plaintext and cryptogram.

The encrypted data items can thus be stored in the same fields/tables as the plaintext. The encryption is done when the data leaves the security perimeter, i.e. before it is stored into the cloud.

Order preserving encryption and tokenisation (OPE and OPT) apply encryption in a way that the order relation of the clear data is maintained, and hence range queries are possible on encrypted data [29].

[Diagram:]

[Known uses:]

Consequences:

☞ **Countered threats:**

The pattern counters several threats from the three areas, which were defined in the "Cloud computing risks and threats" list in section 4.2.

Policy and organisational risks:

As regards **governance and control of the database** which is deployed to the cloud, the end user still depends on the cloud provider's ability and goodwill to grant access to the database in the cloud. Nevertheless, the end use can control the access to the plain data and therefore enforce the confidentiality of the data without reliance on the cloud provider.

The pattern does also not exactly solve, but only ease another common problem, related to the **compliance of an application or a service with legal requirements**. In sensitive areas, regulation requires applications to be certified to meet several requirements, usually laid down in a standard. The process of certification is expensive, and has usually to be re-done in its entirety, even for small modifications of the entire system. So the extension of an unmodified existing system with a cryptographic layer, which is accessed over clearly defined interfaces on the database field level, may significantly ease the re-certification of an information system after the deployment of the database to the cloud.

The risk of **lock-in** with a cloud provider remains unchanged with an encrypted database in comparison to an unencrypted database.

Technical risks:

All the technical risks related with data protection are covered by the cloud security pattern "Moving a legacy application's database to the cloud". These include the risks introduced by **isolation failure** and **interface compromise** due to insufficient measures of the cloud provider, as the data protection risks connected to handling of the data by the cloud provider, including **deletion of the data**.

Still, the risk of a broken **availability chain**, where the data is rendered inaccessible because of a failing internet connectivity, cannot be addressed by this cloud security pattern.

While several technical risks related to the confidentiality of the data are countered, **data loss** and **account hijacking** remain also for the encrypted database.

Data protection risks:

Data protection risks are generally covered when they concern the **confidentiality** and the **integrity** of the data—both are protected by the cryptographic scheme. The risks connected with the **availability** of the data are obviously not mitigated by encryption of the data.

5.3 Field 2: User privacy protection and data minimisation

The following two patterns are closely related to each other – i.e. they can be realised with the same cryptographic primitive: anonymous credentials. Both are concerned with data minimisation—with effectively reducing the data which occurs in transactions with online services and clouds.

5.3.1 Pattern 3: Non-identifiable and untrackable use of a cloud service

Name: Non-identifiable and untrackable use of a cloud service

Summary description: This pattern describes the anonymous interaction of a user with different cloud services, or in general, information services on the internet. So much data is created by users just using systems on the internet and in the cloud. It is just the information that someone accessed some service at some instant of time, which can be accumulated and related to other data in ways beyond the control and interest of the individual connected with the data. So the idea is, to just not let the date be created in the first place. A cryptographic mechanism is required facilitating the anonymous use of services on the internet, without generating data which can be linked to the identity of the user. It should be like using a ticket vending machine in the metro station with a few coins, where a service can be used in anonymity.

[Also known as:]

Example: Further examples for the application of the pattern would be in media services, like music streaming websites or mediatheques, where users could listen to music or read in books without being monitored and tracked in their preferences and behaviour.

Context: This pattern is strongly interrelated to the pattern 4 “Minimise exposure of private data during authentication in the cloud”. In this pattern, anonymity is the goal and linkable data is to be completely excluded, while in pattern 4 some information is revealed—but under the complete granular control of the user. Both patterns can reasonably be combined.

Intention: This pattern wants to provide privacy to users in their daily life in the internet. User privacy shall be supported, against the prevailing trends of ubiquitous surveillance of digital citizens.

Problem: In most current solutions[43], users which have to prove to a verifier that they are authorised to use a cloud system, have to reveal their identity. This generates data which is attributable to individuals and will be accumulated by parties beyond the control of the user for further exploitation, use and potential misuse. In e-government open data applications it may for example be desirable to provide information to eligible users—but not to be able to record who exactly accessed which information. Without specific cryptographic mechanisms, anonymity and unlinkability of transactions cannot be enforced by the user. The user has to rely that the data is not illegally exploited and commercialised—but the mere fact, that the attributable metadata is generated, and appears in log files and during transition in the networks, makes it likely that it is stored and processed beyond any control by the end user.

[43] A illustrative description of current authentication technologies, including anonymous credentials, can be found on IBM’s idemix homepage: http://www.zurich.ibm.com/idemix/whatitdoes.html

More advanced current solutions use an online issuer, who is contacted during each authentication process and who effectively isolates the user from the verifier, so that the actual information, upon which the authentication is based, is not revealed to the verifier. Such a construction is effective in preventing the information flow from the user to the verifier—but as the online issuer has the full knowledge about the transaction and the data involved, and thus constitutes a privacy bottleneck.

Other systems, like ITU-T's X.509 identity certificates, involve an offline issuer, who needs not to be involved during the authentication process and thus does not learn about the single transactions. But in this case the user usually has to reveal his/her entire identity certificate with all the information in it—which is usually not required for the authentication process per se—and what's more, the different transactions of the user become linkable and generate metadata on peoples' behaviour and whereabouts, completely irrelevant for the authorisation of a user for an application or service. The exploitation of such metadata by specialised companies and authorities, and other parties, poses a severe threat against user privacy.

Solution: The PRISMACLOUD project proposes the use of anonymous credentials [30]in cloud systems as a toolbox for the benefit of end users, with the help of which cloud applications and services can be designed for which the user can prove authentication and authorisation without revealing more identity information than necessary.

☞ **PRISMACLOUD primitive:**

Use the technology of 'anonymous credentials' [30] to enable the implementation of privacy protecting and data minimising authentication and authorisation systems for cloud applications and services.

Implications:

- Users may prove the authorisation for a service without revealing their identity;
- Anonymous credentials allow the encoding of attributes in credentials such, that statements about the encoded attributes can be proven to a verifier without revealing the values of the attributes;
- Anonymous credentials are effective tools for data minimisation—the amount of data which is revealed during transactions is effectively reduced;
- Different credential shows can are unlinkable or can be implemented to be unlinkable
- If events need to be linkable, anonymous credentials allow to anonymously prove the possession of a pseudonym

[Diagram:]

Known uses: Non-identifiable and untrackable use of a cloud service is already available in commercial products: **U-Prove** of Microsoft is based on the technology acquired from Credentia.com, the company of the inventor of the zero knowledge proof protocols [31]. IBM's **Identity Mixer** is a "cryptographic protocol suite for privacy-preserving authentication and transfer of certified attributes"[44].

[44] http://www.zurich.ibm.com/idemix/whatitdoes.html

It was “was tested, piloted, improved, and considerably extended throughout (…) participation in a number of European research consortia”[45].

Consequences: “Non-identifiable and untrackable use of a cloud service” is an effective tool for proving the authorisation for a service without revealing the identity of the user. Different credential shows are unlinkable, or only linkable in a defined context, e.g. in a billing application, where the single payments shall be attributed to a user or to a pseudonym—if that is desired.

The pattern can bring tremendous benefits to end users in comparison to cloud applications and services with full identification and user tracking (probably also over different services). It minimises the traces of online activities and thus reduces the risk, that data related to individuals is exploited and commercialised beyond the control of the user.

On the other hand, there are entire businesses living on the identification and tracking of users, which gives them valuable data sets about identifiable end users, e.g. sociological data (who is in contact with whom, and when, data about relationships,), data about habits of all kinds etc. which allows inference of information on individuals far beyond any interest of the end user—data, which e.g. under the lax user privacy regulation in the U.S. of America may be commercialised at discretion of the data collector, and not of the subject related to the data. Some businesses provide services to end users, for which the end users “pay” by granting exclusive usage rights on data, including further processing (“big data”), exploitation and commercialisation in any imaginable way. Such usage patterns are orthogonal to the “Non-identifiable and untrackable use of a cloud service” pattern.

☞ **Countered threats:**

While **policy and organisational risks** do not apply to this current pattern, most important **technical risks** can also be excluded because of the cryptographic security of the primitives which are used for its implementation.

The pattern is very effective in countering **data protection risks**. It allows a fine grained control of which data is exposed to whom. Thus it reduces the risks connected to the information on processing, as well as the risks connected to the loss of control.

[45] http://www.zurich.ibm.com/idemix/eu_projects.html

5.3.2 Pattern 4: Minimise exposure of private data during authentication in the cloud

Name: Minimise exposure of private data during authentication in the cloud

Summary description: Only expose the minimum necessary amount of data when authenticating for a cloud service.

During the process of authenticating, e.g. for accessing a cloud service, a user wants to present some attributes, without revealing other attribute he or she may additionally have. The user may also only want to prove the possession of an attribute, or some quality of the attribute (e.g. a statement on a range it is in) without revealing the exact value of the attribute. The user may want to show or prove attributes to different sites in a manner, that the single showings cannot be linked to the same person.

[Also known as:]

[Example:]

Context: This pattern is strongly interrelated to the pattern 3 “Minimise exposure of private data during authentication in the cloud”. In this pattern, some information is revealed—but under the complete granular control of the user—while in pattern 3 anonymity is the goal and linkable data is to be completely excluded. Both patterns can reasonably be combined.

Intention: The pattern wants to reduce the data which is unnecessarily exposed during authentication situations

Problem: In currents cloud systems, users often reveal much more data than necessary for performing or delegating a specific task. Such data is prone to being accumulated and data mined by the cloud provider and by other parties eventually getting in possession of the data. This represents a severe privacy threat for the user. Currently, authentication for a service in the cloud is often performed by the use of an identity certificate. The user shows the certificate to the verifier who verifies the digital signature on the certificate with the public key of a certifier. The verifier thus learns all the data contained in the identity certificate, although for a proper authentication it might be sufficient to access only a small subset of the data in the identity certificate. Identity certificates also make interactions attributable to the bearer of the identity certificate—i.e. interactions can be tracked across services. All these side effects are problematic from a privacy point of view and data minimisation actually calls for avoiding such unnecessary revelation of data in information infrastructure transactions.

Solution:

☞ **PRISMACLOUD primitive:**

As in Pattern 3 “Non-identifiable and untrackable use of a cloud service”, this pattern uses the technology of “anonymous credentials” [30]. See “PRISMACLOUD primitive” section in Pattern 3.

[Diagram:]

Known uses: Idemix by IBM and **U-Prove** use the technologies of anonymous credentials. For details see the description on the known uses section of the previous Pattern 3 “Non-identifiable and untrackable use of a cloud service”.

Consequences:

The pattern “Minimise exposure of private data during authentication in the cloud” allows an effective reduction of the amount of data which is revealed during authentication and other transactions requiring the presentation of user data.

The pattern enables that statements about the encoded attributes can be proven to a verifier without revealing the values of the attributes. The pattern enables that different credential shows can are unlinkable or can be implemented to be unlinkable. If events need to be linkable, it is possible to anonymously prove the possession of a pseudonym.

☞ **Countered threats:**

As in the related pattern 3, **policy and organisational risks** do not apply to this current pattern, while most important **technical risks** can be excluded because of the cryptographic security of the primitives which are used for its implementation.

The current pattern is also very effective in countering **data protection risks**. It allows a fine grained control of which data is exposed to whom. It thus reduces risks connected to the processing of personal data which is collected by the service provider without effective necessity for the service. It reduces the lack of transparency, and all the risks involved by chain processing involving multiple processors, by moving data between jurisdictions, especially also out of the control of local data protection

5.3.3 Pattern 5: Big data anonymisation

Name: Big data anonymisation

NOTE: This pattern has not yet been defined

Summary description:

[Also known as:]

[Example:]

Context:

Intention:

Problem: Efficient and practical solutions for anonymisation of very big data sets do not exist. K-anonymisation of data [32], which means, that in a set of data, for each entry, there are at least (k-1) other entries, from which it cannot be distinguished, is a NP hard problem [33].

Solution: New, more efficient approaches to anonymising big sets of data have improved in efficiency and are now capable of anonymising very large data sets.

☞ **PRISMACLOUD primitive:**

Primitive 5: Big data anonymisation

[Diagram:]

[Known uses:]

Consequences:

☞ **Countered threats:**

5.4 Field 3: Authentication of stored and processed data

5.4.1 Pattern 6: Protect the authenticity of a data set and possible subsets

Name: Generate an authentic subset from an authentic and signed set of data

Summary description: Subsequently cloaking or/ removing information from an authentic data set, e.g. electronically signed data structure, will be needed to protect the confidentiality of the information that got removed. The aim is to allow and control the removal of some data such that it has a minimal impact on the authenticity of the remaining data. Following the definitions of authenticity protection the technical mechanisms usually require to detect any subsequent modification to a protected data structure as a breach of integrity. At the same time the value of data is increased if the data's origin is authenticatable, such that the originating party cannot technically repudiate having created the data. This pattern gives necessary protection from the time data is created against malicious changes on the message level (not only communication level) for end-to-end communication. It allows to for data minimisation by removing data at a later time not a-priory known. Note, that the removal must be done such that the removed data's confidentiality is protected. Integrity and authentication of origin are valuable properties to retain on data as they are often required to assess the amount of trust that can be placed into the veracity of the information, e.g. knowing it comes from a trustworthy source and has not been changed in unauthorised ways induces trust in the suitability of the information encoded in the data for the task at hand. Some processes even require the level of integrity and authentication of origin protection to comply with the legal requirements (e.g. qualified electronic signatures as regulated in eIDAS) to increase the legal value of evidence of the data. At the same time subsequent to the integrity protection information must be changed to allow to comply with privacy and data protection requirements. This pattern's change is the complete removal of data. Any change must be done securely, meaning that no information from the removed data can be gained from looking at the remaining data's integrity and authenticity protection.

Also known as: Problem: Document Sanitization Problem [34]; Cryptographic solutions: Content Extraction Signature [35], Homomorphic Signature Scheme [36], Redactable Signature [37] [38], Malleable Signature [39], Sanitizable Signature[46]

Example: Medical data is the most prominent example, hence we use an example from that domain as well. Imagine a number of tests are carried out on a blood sample and a report is being created. For this example imagine the blood test report would contain information about the levels of seven (in reality there are more) indicators: (1) fasting glucose (blood sugar), (2) total cholesterol, (3) haemoglobin (Hgb), (4) ALT (alanine aminotransferase), (5) thyroid, (6) vitamin D, (7) tuberculosis (TB).

In the context of the example the document sanitization problem would manifest as the problem to remove the actual test results, e.g. only the results showing blood sugar, cholesterol and vitamin D (tests 1, 2, and 6) shall be given to your ecotrophologist. However the values that are given shall be protected against malicious tampering ever since they left the credible source and the source shall

[46] Only a few papers call the authorisation of only the removal of data items "sanitization", e.g. [55]. In PRISMACLOUD, we adopted the concept for sanitizable signatures that was introduced by Ateniese et al. also in 2005 [41] which was also adopted in many other recent and influential papers and which has been shown to be cryptographically different to constructions that allow just the removal [56].

be verifiable. The credible source is the doctor/laboratory/machine that did the tests to obtain the results. The party subsequently modifying it is the patient. The party that wants to assess the credibility is yet another party, e.g. the ecotrophologist. Hence, even after the removal, the remaining test results shall give the same credibility as if only those remaining results were done on its own. There are several details or sub-problems to the document sanitization problem that we want to briefly highlight: Does the cloaking reveal that cloaking has taken place? Does it reveal where cloaking has been done? For example the ecotrophologist shall not know that you have been tested for tuberculosis (test 7), as this test is only done if you are in a high risk group or are already receiving treatments. As the mere knowledge that this test was done can be regarded as personally indiscriminating information the subsequent cloaking shall perfectly remove this information. However, if a certain test is necessary to be performed, e.g. is required to meet due diligence procedures, it needs to be recorded as having been carried out. This might also help to provide evidence in order to charge the insurance the costs of the test, then the actual result shall be cloaked in such a way that it is still verifiable that initially a value was inserted, i.e. the test was actually done as some result has been obtained. To stay with a visual example, it shall not be possible to cloak empty values such that they can pose as test results. For the undermining cryptographic mechanisms and their application this means that many mathematical and implementation details of the cryptographic tools need to be tailored and fine-tuned correctly to give the desired results.

Context: This pattern is applicable whenever data originates at a credible source and the data's trustworthiness depends on the source being authenticatable and the data being subjected only to benign, i.e. authorised, subsequent modifications. It can applied whenever data is integrity protected by signatures: Using a mechanisms that detects the absence of authorised and unauthorised changes applying malleable signatures allows the same protection as a standard signature, however if a purposeful verification can be done on a subset of the information, i.e. the protective scope of the integrity protection must change for reasons of privacy/data protection/trade secret protection, then this pattern allows the scope to be subsequently adjusted.

Intention: The application of this pattern allows to cater for future subsequent removal of data from a data set for which integrity and authenticity protection mechanisms such as digital signatures are usually applied. Future in this context means that at the time of protection the needed combinations of removals is not yet known. The pattern allows for the remaining data from the data set:

- to detect the absence of any unauthorised change on the remaining data (including the actual content and the data's structure or relations within the structure),
- to authenticate the origin of the data set via a cryptographic key[47],
- to provide non-repudiation of generation for the remaining data;

while at the same time

- the removed data's confidentiality is protected.

Problem: Currently well accepted and widely used standard digital signatures have the drawback that once the verification of the integrity check value fails, the integrity and authenticity protection

[47]e.g. with digital signatures the successful verification with the public signature verification key establishes evidence that the signature was created with the knowledge of the secret signing key

for the such-protected document as a whole is invalidated. Any subsequent edit of the data, authorised or not, will get detected but with the consequence that the integrity can no longer be established for any of the remaining data. Naïve and obvious solutions to the integrity problem exists, e.g. hash-trees, but suffer from not offering privacy with a cryptographic sufficient strength. A standard hash still mathematically depends on all the input, thus removing some input does not allow to remove this information from the hash and thus leaves this cryptographically not offer a sophisticated level of privacy ([40]).

Solution: Employ a different set of cryptographic primitives or in a different combination than conventional digital signature schemes such that the malleability is enabled while authenticity for the remaining data and confidentiality of the removed data can be achieved.

☞ **PRISMACLOUD primitive:** PRISMACLOUD wants to employ the technology of malleable signatures (cf. [32]) which allow controlled modifications (here we require deletion or redaction) of certain parts of the signed data without the signature losing its validity. The allowed modifications are being formally described and the malleable signature for a specific data item is created. At a later time the authenticity of the modified data can be verified, and thus, the verifying entity can gain cryptographic assurance about the remaining data's origin and that only allowed modifications were made.

[Diagram:]

[Known uses:]

Consequences: The application of the pattern "Generate an authentic subset from an authentic/signed set of data" allows to apply cryptographic integrity protection at an early point in the data generation such that the origin of data and the absence of modifications can be verified at any later time by third parties but it still allows the flexibility to remove data later to apply the need to know principle when such protected data is to be disseminated. Thus it combines the strength of cryptographic end-to-end integrity protection with ability to remove data to do data minimisation.

☞ **Countered threats:**

The application of this pattern counters at least the following four threats:

- Loss of data integrity: The remaining data is still integrity protected, any change to data that has not been removed will be detected as with standard digital signatures or other integrity protection mechanisms.
- Loss of accountability: The remaining data's origin can still be authenticated by the public key that is used for digital signature verification. Within the limits of the pattern the actual strength of the cryptographic algorithm could be tailored to achieve different levels of technical and with it legal assurance (as discussed in D2.1 malleable signatures are technically as strong as qualified electronic signatures which allow to assign high evidentiary value to documents).
- Data leakage: Data that might not be needed by all parties can be marked as removable at the time of signature generation. Thus, while preserving full authenticity protection (integrity + authentication of origin) just the remaining data – the data needed to fulfil an action– can to be given to the requesting party. If this pattern was applied during the

generation of the signature the unneeded data can be removed from the set without having any impact on the integrity and authenticity protection.

- Insecure or incomplete data deletion: This pattern allows that if the data that is requested to be removed is contained in a data structure that was protected for integrity and is of authentic origin, the removal of that data will not invalidate the remaining data structure. Thus, it allows to delete data if requested without the need to recreate the protection for integrity and authenticity. This removes a potential obstacle that might have caused hesitation to delete data at all occurrences.

5.4.2 Pattern 7: Authorise controlled subsequent modifications of signed data

Name: Authorise controlled subsequent modifications of signed data

Summary description: One practical advantage of cloud systems is that collaborative applications may easily be implemented. In order to control the authenticity of data that are passed between applications, current solutions use electronic signatures, but those have at least the following drawbacks:

- In collaborative applications, several parties usually also need to modify common data;
- Common electronic signatures are static: one single modification in the authenticated data structure invalidates the signature and removes the authenticity property from the whole data structure.

Also known as: Sanitizable signatures (conceptually described in [41]), blanket signatures [42]

Example: Assume that an invoice is generated as a structured document, containing a couple of fields that are filled with information by other parties; e.g. the invoice for your data center usage this month depends on the resources consumed in the storage services cloud and the compute services cloud and hence those two infrastructure providers are being asked to fill into your invoice the exact amount of service used in this month, then the accountant of the data center will multiply the service consumption by the service charge and calculate the total. The invoice can thus be split into five parts, two parts for each service's consumption, two for the service charge of each service and one for the total amount. Now one workflow to generate the invoice that is signed by your semi trusted data center provider is that the accountant fills in only the service charges as agreed by contract and marks them as non editable, but the fields for the actual consumptions he marks as being filled in by the compute and storage service provider respectively, and finally the total he marks as editable by himself. Now the customer can check that empty invoice to identify that the service charges are correct (he might even sign them with the same fields marked as edible by the same parties). Then the two services add the actual consumption values which only they can do without invalidating the signature as they have been authorised to do so. As the final step the accountant of the data center calculates the total and if the signature on the invoice after the service consumption updates is still valid updates the last part for the total which only he can update without invalidating the signature. The customer can now validate the signature to see that only parties that the data center authorised have been modified. Here the customer might not need to know or trust the services. If the customer counter signs he must specify exactly which parties are able to modify the data without invalidating the signature, thus the customer can pin point the services.

Another application domain is the need to change data in order to make it less specific, e.g. dilute it to reduce data quality but increase data privacy. If this needs to be done to signed data, than this pattern can help.

Context: Firstly, note that PRISMACLOUD acknowledges that this pattern is closely related to a pattern that might be known as "delegation". However, the primitive is in details different from delegatable signatures. It is yet on PRISMACLOUD's future work to revise this pattern and make sure that the details that differentiate them are explicitly stated.

Secondly, this pattern is closely related to Pattern 6 "Subsequently cloaking /or removing information from electronically authentic electronic documents with minimal impact on the authenticity signed

of the remaining information" Specifying that another party can subsequently adjust / modify a document in an defined way. Allowing controlled/confined subsequent changes allows those subsequent actions to be seen as delegations.

Intention: The application of this pattern allows to cater for future subsequent modifications of data for which integrity and authenticity protection mechanisms such as digital signatures are usually applied. Future in this context means that at the time of initial application of the protection the modification or the party allowed to do it might not yet be known. The pattern allows that it is possible for the resulting modified data as a whole:

- to detect the absence of any unauthorised modification to the data (including the actual content and the data's structure or relations within the structure),
- to authenticate the origin of the data set via a cryptographic key ,
- to provide non-repudiation of generation for the data that was not subject to modifications;

while at the same time

- the data's previous state, i.e. before the modification, remains confidential.

Problem: Currently well accepted and widely used standard digital signatures have the drawback that once the verification of the integrity check value fails, the integrity and authenticity protection for the such-protected document as a whole is invalidated. Any subsequent edit of the data, authorised or not, will get detected but with the consequence that the integrity can no longer be established for any of the remaining data. Additionally, not only shall the authorised change be not harmful to the integrity protection and the accountability of the remaining unchanged data, but the modified data shall protect the confidentiality of the previous version of the data (i.e. before the change). Further, it might be necessary that not only the fact that the party that changed it was authorised to do so, but maybe accountability of the party that did an actual change might be needed. In all cases a main requirement is that data can be modified only in an authorised way by only authorised parties without the need to re-apply the originating parties signature (e.g. without the secret signature generation key being needed for resigning the data after the modification, which would be a naïve solution to the problem).

The property of accountability might be needed in different fashions. D4.4 already identified the following:

- non-interactive and public accountability would allow all verifiers to check accountability with just needing the public keys and no interaction (e.g. like digital signatures)
- interactive accountability would require the party accused of being accountable to adhere to a rebuttal protocol (which involves the parties secret) to generate a proof that shows that they are indeed not accountable (e.g. party A signs and authorises B, B has changed the data in an authorised way and now accuses A to be accountable for the changed data; then party A can produce a proof showing that it was changed by the authorised party B)

For both the accountability could be on the whole data structure (e.g. if B changed one data entry in a protected data structure it becomes accountable for the whole data structure) or on the admissible

parts (e.g. if B changed part number 7 than B can be found accountable for the part number 7, but not for number 5 unless B changed it).

PRISMACLOUD in D4.4 has already defined those properties cryptographically [43].

Solution: Employ a different set of cryptographic primitives or in a different combination than conventional digital signature schemes such that the malleability is enabled while authenticity for the remaining data and confidentiality of the overwritten original version of the data can be achieved.

☞ **PRISMACLOUD primitive:** employ the technology of sanitizable signatures or functional signatures (the overview of the state of the art in cryptography is in D4.4 [43]) which allows controlled modification (replacing substrings in case of sanitizable signatures or computing the function for functional signatures) of certain parts of the signed data without the signature losing its validity. The allowed modifications are being formally described and the malleable signature for a specific data item is created. At a later time the authenticity of the modified data can be verified, and thus, the verifying entity can gain cryptographic assurance that only allowed modifications were made only by allowed entities.

Implications:

1. Only controlled modification is allowed on the data;
2. Allowed modifications do not need the secret signing key;
3. Modification may be allowed for everyone, or for specific parties in possession of a specific cryptographic key;
4. Correct modification preserves the validity of the signature;
5. Modification beyond what is allowed, renders the signature invalid. The authenticity property for the entire signed data item is destroyed;
6. Allowed modifications may be described on a document level (which parts may be edited by substituting the existing string of bits with an arbitrary new string of bits that could be longer or shorter) or allow the application of specific arithmetic functions (see point 7);
7. As an arithmetic function that limits the modification, only linear functions (counting, summation…) and polynomial functions (variance, covariance…) are feasible;
8. Arbitrary functions are possible in theory, but currently not practically feasible.

More technical details on the cryptographic underpinnings and existing cryptographic methods can be found in PRISMACLOUD Deliverable D4.4.

[Diagram:]

[Known uses:]

Consequences: The application of the pattern "Authorise controlled subsequent modifications of signed data" allows to apply cryptographic integrity protection at an early point in the data generation such that the origin of data and the absence of modifications can be verified at any later time by third parties but it still allows the flexibility to modify this data, to allow later computations or necessary changes. As the authorisation is cryptographically protected the absence of violations, i.e. that no unauthorised overstepping of authorised limits has occurred, are verifiable on the data especially after the modification. This allows to use it to pre-define computing or workflows. Still, different freedom might be given what a modification is, thus if data later requires to be sanitized to minimise the data it can be done without any additional interaction with the original signer, if the party has been authorised. Thus it combines the strength of cryptographic end-to-end integrity protection with ability to authorise subsequent computation. The term "computing on authenticated data" was used in [44] to describe this concept.

☞ **Countered threats:** The application of this pattern counters at least the following four threats:

- Loss of data integrity: The modified data is still integrity protected, any change to data that has not been authorised by the initial signer will be detected as with standard digital signatures or other integrity protection mechanisms.
- Loss of accountability: The remaining data's origin can still be authenticated by the public key that is used for digital signature verification. Within the limits of the pattern the actual strength of the cryptographic algorithm could be tailored to achieve different levels of technical protection (i.e. accountability can be interactive or non-interactive, or on the whole data or for each individual part of the data) and with it different legal assurance (as discussed in D2.1 malleable signatures are technically as strong as qualified electronic signatures which allow to assign high evidentiary value to documents).
- Wrong operation: If the authorisation encoded in the malleable signature defines the expected workflow or the allowed computations than a failed verification of the data's integrity via the signature verification indicates a failure. This allows could users to regain the possibility to generate evidence that the Cloud service did not perform as expected.
- Data leakage: Data that might need to be overwritten and previous data must not be accessible by parties that do not know previous versions. Thus, while preserving full authenticity protection (integrity + authentication of origin) the recent, modified version can be given to the requesting party. If this pattern was applied during the generation of the signature the modifiable part of the data can be modified in a controlled manner by the appointed party without requiring the verifier to learn the unmodified version.

5.4.3 Pattern 8: Controlling the correctness of delegated computations

Name: Controlling the correctness of delegated computations

Summary description: This pattern allows to delegate computations on outsourced data to third parties, such that the data owner and/or other third parties can verify that the outcome has been computed correctly. The verification shall be efficient, such that the verification involves much lower computational costs than that required to perform the computation unilaterally. Further, the verification shall give a cryptographically credible proof that the computation was right or wrong to serve as evidence for showing either that everything was done correctly (e.g. to prove having fulfilled your duty or due diligence) or to be used as an evidence for accusation (e.g. to be the tool that cloud users need to identify and prove that the cloud provider broke the SLA). In more detail, the pattern might need to be refined to cater for additional properties such differentiating if there is a need for providing privacy for the inputs while the verification of the outputted computed result is still possible or also which functions are to be computed

Also known as: Verifiable Computing

Example: Note that cloud providers storing data or performing computations on them cannot be considered fully trustworthy or immune to attacks. Thus, a very important and relevant research question is how one can outsource data and computations to a non-trusted third party such that this party can process the data and at the same time provide guarantees that integrity and if needed confidentiality of inputs (or inputs and outputs) has been preserved. Assume in an eHealth scenario that a medical device generates trusted and authentic readings of the patient's current condition. Assume now that this data is given to the cloud to analyse it. For merits of simplicity in this example, and because it is already a very valuable and valid application scenario, assume that the data is recorded in very small time intervals, e.g. 1 reading per minute and the computation shall yield the daily average. The average shall be verifiable to be computed correctly. The example is overly simplified, but allows to highlight the property of input confidentiality: Just assume the average must remain verifiably correct without having access to the input data as this would allow to infer daily behavioural patterns that the averaging would hide (e.g. does the patient do enough workout during the day could be required to be checked, but not if he does workout in the morning and in the afternoon which might allow to deduce that he was not at work).

Context: This pattern is relevant whenever cloud providers are performing computations on data but cannot be considered fully trustworthy or immune to attacks on the integrity. Thus, it is complementary to the patterns that have been described for encrypted storage or encrypted computations, both having the main goal to guarantee confidentiality and integrity, but for storage (e.g. data at rest and in transit).

Intention: This pattern shall be used whenever data is transferred for computation to a third party in the cloud, but the party receiving the data cannot process (run the computation) the data and at the same time provide a sufficient guarantee that integrity (and confidentiality if needed) has been preserved.

Problem: The delegation of computing cannot be verified without a dedicated application serving all involved parties (outsourcer, cloud provider, verifier). No solution is available that provides both the

security and privacy level needed for sensitive data and the flexibility and efficiency to be used in practice. One example for such sensitive data are electronic health records as generated and processed in the PRISMACLOUD eHealth scenario.

Solution:

☞ **PRISMACLOUD primitive:**

Proposed solution: In verifiable computing [45], a client hands data to a cloud service and if the client requests the computation of some (arbitrary) function over the data, the cloud service returns the result of the computation together with a proof (this process may also be interactive). By means of the proof, the client can (efficiently) decide whether the requested function has been correctly applied to the outsourced data. While general purpose solutions to verifiable computing are quite inefficient, we will focus on the computation of some limited class of functions, which can be quite efficiently be realized by using for instance malleable (homomorphic) signatures.

[Diagram:]

[Known uses:]

Consequences:

1. Verifiable computing allows new types of collaborative applications
2. Efficient solutions are only available for simple calculations (linear functions, e.g. sums)
3. The privacy of the outsourced data is typically not regarded. But the concept of verifiable computations can also be used to certify that the cloud provider has not conducted certain privacy-invasive operations, such as profiling operations. In this sense, privacy can be promoted.

☞ **Countered threats:**

The pattern removes the trust in the cloud provider to do the correct calculation at the cost that one has to run potentially costly verify algorithms.

The pattern also provides evidence that the calculations were correct or incorrect.

5.5 Field 4: Certification of virtualised infrastructures

5.5.1 Pattern 9: Controlling your virtual infrastructures

Name: Controlling your virtual infrastructures

Summary description: A cloud user has rented a virtual infrastructure which is hosted by a cloud provider. The cloud user can verify that the rented virtual cloud infrastructure is properly configured at the cloud provider. The cloud user can effectively check the proper isolation of the rented cloud infrastructure.

[Also known as:]

[Example:]

Context: The pattern applies to situations where a customer or end user rents a virtual infrastructure from a cloud service provider. The underlying NIST service model is IaaS, "Infrastructure as a Service". The nature of the IaaS cloud service model is, that the rented infrastructure is physically hosted in the data center of a cloud provider, where the rented infrastructure is virtualised on an array of computers with the help of a hypervisor software.

Intention: Users want to have a means of control that the infrastructure they have rented from a cloud provider is securely configured. Users want to have assurance that their virtualised infrastructure is properly isolated from other tenants hosted "in the vicinity" (i.e. on the same machine, by the same hypervisor).

Problem: End users who rent an infrastructure from a cloud provider have to rely on the cloud provider that the virtual infrastructure is properly configured. It is by certification according to a recognised standard that a cloud provider wants to increase the trust of the customer, i.e. the end user. Technically, there exist measures for the attestation of the security of physical and virtual machines. Trusted components monitor the systems on all levels and layers.

Solution: An auditor (this can be a human auditor, or a machine) verifies an actual infrastructure, represents it as a graph, and signs the graph with an electronic graph signature scheme. With the help of this the resulting graph signature, the verification of the auditor is bound to the actual infrastructure as it was configured at the time when the audit was carried out.

The graph signature algorithm lets the customer prove topology properties of the virtualised infrastructure (like connectivity isolation) without revealing to the customer actual details of the topology.

☞ **PRISMACLOUD primitive:**

Using recently developed methods [46] for representing virtualised infrastructure as graphs, i.e., a set of nodes interconnected by edges, a cloud topology signature scheme can be implemented.

[Diagram:]

[Known uses:]

Consequences: With the help of this the resulting graph signature, the verification of the auditor is bound to the actual infrastructure as it was configured at the time when the audit was carried out.

The end user has a means in hand to verify the state of a cloud configuration, as it was at the time of a previous audit.

The graph signature algorithm framework lets the customer prove topology properties of the virtualised infrastructure (like connectivity isolation) without revealing to the customer actual details of the topology.

☞ **Countered threats:**

The pattern "Controlling your virtual infrastructures" is mainly addressing technical risks, being **isolation failure** and several other items from the technical risks which have to do with the configuration (or misconfiguration) of a rented cloud infrastructure.

Proposed solution: Using recently developed methods for representing virtualised infrastructure in graph structures [46], extend current audit procedures with a means for proving the correct configuration of virtualised infrastructures.

Implications:

1. A (human) auditor verifies an actual infrastructure and represents it in a graph, which he signs with a graph signature. With the help of this graph signature, the verification of the auditor is bound to the actual infrastructure as it was configured at the time when the audit was carried out.
2. The graph signature algorithm lets the customer prove topology properties of the virtualised infrastructure (like connectivity isolation) without revealing to the customer actual details of the topology.

5.6 Outlook

In this chapter we introduced the idea of developing design patterns for the security and privacy functionalities yielded by our PRISMACLOUD cryptographic cloud security primitives. The idea is to jumpstart the use of the design pattern methodology, within the project and beyond it, to describe the usefulness and the areas of use for the foreseen cryptographic mechanisms for a broad audience—from the scientists, who do the cryptographic research and develop the cryptographic primitives, to the application developers, to the cloud service designers and cloud providers, to the end users who use the cryptographic primitives in applications and services.

We intend to further use and develop this notion of cloud security patterns in project deliverables, e.g. of WP7 "Composition of next-generation secure cloud services", in T 7.1 "Security and privacy by design", and in the "holistic security models" of T7.2, in support of T7.3 "Architecture and guidelines for secure service composition", as well as in the software development process of T7.5.

6 List of Figures

7 List of Tables

8 Abbreviations and acronyms

.NET	.NET software framework (by Microsoft)
AES-256	Advanced Encryption Standard
ALT	alanine amino transferase
API	Application Programming Interface
authN	Authentication
authZ	Authorisation
AWS	Amazon Web Services
CaaS	Communication as a Service
CAPEX	Capital Expenditures
Cloud OS	Cloud Operating System
CloudHSM	Cloud Service by Amazon (Cloud – Hardware Security Module)
CP	Cloud Provider
CPU	Central Processing Unit
CSA	Cloud security Alliance
DaaS	Data as a Service
DDoS	Distributed Denial of Service attack
EC2	Elastic Computing Cloud (Amazon)
EEA	European Economic Area
eIDAS	electronic IDentification and Authentication Services - The new European Regulation on Electronic Identification and Trust Services for Electronic Transactions in the Internal Market[48]
ENISA	European Union Agency for Network and Information Security
FIPS	Federal Information Processing Standard (USA)
FISMA	Federal Information Security Management Act of 2002 (USA)
FP7	7th Framework Program of the EC
FPE	Format Preserving Encryption
FPT	Format Preserving Tokenisation
GAE	Google App Engine
GPU	Graphics Processing Unit
GSI	Grid Security Infrastructure
HaaS	Hardware as a Service
HCI	Human Computer Interaction
HDInsight	A Microsoft product
Hgb	Haemoglobin
HIPAA	Health Insurance Portability and Accountability Act of 1996 (USA)
HITECH	The Health Information Technology for Economic and Clinical Health Act (USA)
HPC	Federal Information Security Management Act of 2002 (USA)
HSM	Hardware Security Module
HW	Hardware
IaaS	Infrastructure as a Service
IAM	Identity and Access Management
ICDES	IBM Cloud Data Encryption Services

[48] European General Data Protection Regulation (GDPR), i.e. Regulation (EU) No 910/2014; most of it will take effect from 1 July 2016,
online: http://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32014R0910&from=EN

ICT	Information and Communications Technology
IDS	Intrusion Detection System
Interoute VDC	Interoute Virtual Data Center – A product by Interoute
IP	Internet Protocol
IPS	Intrusion Prevention System
IPSec	Internet Protocol Security
IRT	Interoute
IT	Information Technology
ITU	International Telecommunication Union
ITU-T	ITU Telecommunication Standardization Sector
MaaS	Monitoring as a Service
MAC	Message Authentication Code
MPLS/IP	Multiprotocol Label Switching / Internet Protocol
NIST	National Institute for Standards and Technologies (USA)
NP	Nondeterministic Polynomial, a computational complexity class
OPE	Order Preserving Encryption
OPT	Order Preserving Tokenisation
OS	Operating System
PaaS	Platform as a Service
PBX	Private Branch Exchange
PCI	Payment Card Industry
PCI DSS	PCI Data Security Standard
PingID	a proprietary multi-factor authentication software
PKI	Public Key Infrastructure
PLoP	Pattern Languages of Programs, a Conference
QoE	Quality of Experience
S3	Amazon S3 – Simple Storage Service
SaaS	Software as a Service
SAS70	Statement on Auditing Standards No. 70: Service Organizations (of the American Institute of Certified Public Accountants (AICPA)
SDK	Software Development Kit
SIEM	Security Information and Event Management
SQL	Structured Query Language
SSO	Standards Settion Organisation
UML	Unified Modeling Language
VDC	Virtual Data Center
VLAN	Virtual Local Area Network
VPC	Virtual Private Cloud
VPN	Virtual Private Network
VXLAN	Virtual Extensible Local Area Network
WAN	Wide Area Network
XaaS	Anything as a Service
XSS	Cross-Site Scripting

9 References

(NOTE: All online references were accessed in Dec 2015)

[1] European Commission, “European Cloud Computing Strategy: Unleashing the Potential of Cloud Computing in Europe,” *COM(2012) 529 final; (online: http://ec.europa.eu/digital-agenda/en/european-cloud-computing-strategy),* 27 September 2012.

[2] P. Mell and T. Grance, “The NIST Definition of Cloud Computing,” *NIST Special Publication 800-145 (online: http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf),* September 2011.

[3] L. Badger, T. Grance, R. Patt-Corner and J. Voas, “Cloud Computing Synopsis and Recommendations,” *NIST Special Publication 800-146 (online: http://csrc.nist.gov/publications/drafts/800-146/Draft-NIST-SP800-146.pdf),* 2012.

[4] P. Mell and T. Grance, “The NIST Definition of Cloud Computing,” 2011.

[5] P. Costa , M. Migliavacca, P. Pietzuch and A. L. Wolf, “NaaS: Network-as-a-Service in the Cloud,” *USENIX, Hot-ICE '12 San Jose, CA (online: https://www.usenix.org/system/files/conference/hot-ice12/hotice12-final29.pdf),* 2012.

[6] H. Li, C. Spence, R. Armstrong, R. Godfrey, R. Schneider, J. Smith and R. White, “Intel Cloud Computing Taxonomy and Ecosystem Analysis,” Intel Information Technology, 2010.

[7] C. Karagiannis and G. Höfer, “Cloud computing services: taxonomy and comparison,” 2011.

[8] L. Youseff, M. Butrico and D. D. Silva, “Toward a Unified Ontology of Cloud Computing,” IEEE, 2008.

[9] F. Liu, J. Tong, J. Mao, R. Bohn, J. Messina, L. Badger and D. Leaf, NIST CLoud Computing Reference Architecture; Special Publication 500-292, online: http://www.nist.gov/customcf/get_pdf.cfm?pub_id=909505, Recommendations of the National Institute of Standards and Technology.

[10] B. McNee, “Digital Business Rethinking Fundamentals,” 2014.

[11] Q. Ling, L. Zhiguo, D. Yujian and G. and Leitao, “Cloud Computing: An Overview Ling,” Springer Berlin Heidelberg, Bejing, 2009.

[12] N. A. Sultan, “Reaching for the "cloud": How SMEs can manage,” Liverpool, International Journal of Information Management, Volume 31, Issue 3, 2011, p. 272–278.

[13] IBM, “IBM Cloud & Smarter Infrastructure Training,” 2015.

[14] Rackspace US, Inc., “Rackspace Privacy Statement”.

[15] NIST Cloud Computing Standards Roadmap, NIST Special Publication 500-291 Verison 2, July 2013 (Supersedes Version 1.0 of July 2011), NIST Cloud Computing Standards Roadmap Working Group.

[16] "Cloud computing; Benefits, risks and recommendations for information security; Rev. B. December 2012," [Online]. Available: online: https://resilience.enisa.europa.eu/cloud-security-and-resilience/publications/cloud-computing-benefits-risks-and-recommendations-for-information-security.

[17] Article 29 Data Protection Working Party; opinion 05/2012 on Cloud Computing; Adopted July 1st 2012; 01037/12/EN WP 196, online: http://ec.europa.eu/justice/data-protection/article-29/documentation/opinion-recommendation/files/2012/wp196_en.pdf.

[18] CSA, Security guidance for critical areas of focus in cloud computing, Cloud Security Alliance, 2011.

[19] European Data Protection Directive 95/46/EC on the protection of individuals with regard to the processing of personal data and on the free movement of such data, online: via the official EC data protection home page http://ec.europa.eu/justice/data-protection/index_en.htm.

[20] C. Alexander, S. Ishikawa and M. Silverstein, A Pattern Language: Towns, Buildings, Construction, Oxford University Press ISBN 0-19-501919-9., 1977.

[21] E. Gamma, R. Helm, R. Johnson and J. Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley. ISBN 0-201-63361-2, 1994.

[22] M. Schumacher, E. Fernandez-Buglioni, D. Hybertson, F. Buschmann and P. Sommerlad, Security Patterns - Integrating Security and Systems Engineering, West Sussex, England: John Wiley & Sons, Ltd, 2006.

[23] N. Doty and M. Gupta, Privacy Design Patterns and Anti-Patterns, A Turn for the Worse: Trustbusters for User Interfaces Workshop July 24-26, 2013, Newcastle, UK, 2013.

[24] T. Lorünser, A. Happe and D. Slamanig, "ARCHISTAR: Towards Secure and Robust Cloud Based Data Sharing," *IEEE 7th International Conference on Cloud Computing Technology and Science, CloudCom 2015, Vancouver, November 30 - December 3,* 2015.

[25] D. Slamanig and C. Hanser, "On Cloud Storage and the Cloud of Clouds Approach," *ICITST-2012,* pp. 649-655, 2012.

[26] A. Shamir, "How to Share a Secret," *Commun. ACM 11 Vol 22,* pp. 612-613, 1979.

[27] J. Müller-Quade and D. Unruh, "Long-Term Security and Universal Composability," *J. Cryptology,* vol. 23, no. 4, 2010.

[28] "Patent Application 2011/0280394. Format-Preserving Encryption Via Rotating Block Encryptions. https://www.google.com/patents/US20110280394".

[29] A. Boldyreva, N. Chenette, Y. Lee and A. O'Neill, “Order-preserving symmetric encryption.”.*Proceedings of Eurocrypt’09, Volume 5479 of LNCS.*

[30] J. Camenish, A. Lehmann and G. Neven, "Electronic Identities Need Private Credentials," *IEEE Security,* vol. 10, pp. 80-83, 2012.

[31] S. Brands, Rethinking Public Key Infrastructures and Digital Certificates: Building in Privacy, MIT Press, 2000.

[32] L. Sweeney, "K-anonymity: a model for protecting privacy," *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems,* vol. 5, pp. 557-570, 2002.

[33] A. Meyerson and R. Williams, "On the complexity of optimal k-anonymity," *Symposium on Principles of Database Systems, PODS '04, New York, U.S.A.,* 2004.

[34] I. Miyazaki, T. M. M. Iwamura, R. Sasaki, H. Yoshiura, S. Tezuka and Hideki, “Digitally Signed Document Sanitizing Scheme with Disclosure Condition Control,” *IEICE Transactions,* pp. 239-246, 2005.

[35] R. Steinfeld, L. Bull and Y. Zheng, “Content Extraction Signatures,” in *4th International Conference on Information Security and Cryptology (ICISC 2001)*, 2002.

[36] R. Johnson, D. Molnar, D. Song and D. Wagner, “Homomorphic signature schemes,” in *Topics in Cryptology (CT-RSA 2002)*, 2002.

[37] K. Miyazaki, Interviewee, *Redactable Digital Signatures for Secure and Easy-to-use Digital Document Systems.* [Interview]. 21 May 2008.

[38] D. Slamanig and C. Stingl, “Disclosing verifiable partial information of signed CDA documents using generalized redactable signatures,” in *11th International Conference on e-Health Networking, Applications and Services, 2009. Healthcom 2009.*, 2009.

[39] L. Wei, S. E. Coull and M. K. Reiter, “Bounded vector signatures and their applications,” in *Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security*, 2011.

[40] C. Brzuska, H. C. Pöhls and K. Samelin, “Non-Interactive Public Accountability for Sanitizable Signatures,,” *Proc. of the 9th European PKI Workshop: Research and Applications (EuroPKI 2012),* p. 178, 2012.

[41] G. Ateniese, D. H. Chou, B. d. Medeiros and G. Tsudik, “Sanitizable Signatures,” in *Proceedings of the 10th European Symposium on Research in Computer Security (ESORICS 2005)*, 2005.

[42] D. Slamanig and C. Hanser, “Blank Digital Signatures,” in *Proceedings of AsiaCCS 2013*, 2013.

[43] D. Demirel, D. Derler, C. Hanser, H. Pöhls, D. Slamanig and G. Traverso, PRISMACLOUD D4.4: Overview of Functional and Malleable Signature Schemes, 2015.

[44] J. H. Ahn, D. Boneh, J. Camenisch, S. Hohenberger, A. Shelat and B. Waters, “Computing on Authenticated Data,” in *TCC*, 2012.

[45] Walfish, M. and Blumberg, A. J. "Verifying Computations without Reexecuting Them"; Commun ACM, 2015, 58, p.74-84.

[46] T. Groß, "Signatures and Efficient Proofs on Committed Graphs and NP-Statements," *Financial Cryptography,* 2015.

[47] ISO/IEC 27002, Information technology -- Security techniques -- Code of practice for information security controls, Geneva, Switzerland: International Organization for Standardization, 2013.

[48] P. Drucker, “The Age of Discontinuity,” *New York: Harper & Row. ISBN 978-1-56000-618-3,* 1969.

[49] M. Walfish and A. J. Blumenberg, "Verifying Computations without Reexecuting Them," *Commun. ACM,* vol. 58, no. 2, pp. 74-84, 2015.

[50] D. Catalano, "Homomorphic Signatures and Message Authentication Codes," *SCN,* pp. 514-519, 2014.

[51] M. Hafiz, “A collection of privacy design patterns,” *Proceedings of the 13th Pattern Languages of Programs, PLoP 2006,* October 2006.

[52] C. Fehling, F. Leymann, R. Retter, W. Schupek and P. Arbitter, Cloud Computing Patterns, Wien: Springer-Verlag.

[53] T. Erl, Cloud Computing: Concepts, Technology & Architecture, Prentice Hall/PearsonPTR, 2016.

[54] P. Mell and T. Grance, “The NIST Definition of Cloud Computing,” *NIST Special Publication 800-145,* September 2011.

[55] T. Izu, N. Kanaya, M. Takenaka and T. Yoshioka, “PIATS: A Partially Sanitizable Signature Scheme,” in *Information and Communications Security*, 2005.

[56] H. d. Meer, H. C. Pöhls, J. Posegga and K. Samelin, “On the Relation between Redactable and Sanitizable Signature Schemes,” in *ESSOS*, 2014.

[57] A. G. B. Fisher, “Production, primary, secondary and tertiary,” *Economic Record 15.1,* pp. 24-38, 1939.

v1.0b