ALMA MATER STUDIORUM – UNIVERSITÀ DI BOLOGNA
DOTTORATO DI RICERCA IN DIRITTO E NUOVE TECNOLOGIE XXVI° CICLO

The twofold role of Cloud Computing in Digital Forensics: target of investigations and helping hand to evidence analysis

Corrado Federici

Relatore
Prof. Cesare Maioli

Co-Relatore
Prof. Roberto Di Pietro

Coordinatore
Prof. Giovanni Sartor

Esame finale anno 2014

Sigla Settore Concorsuale:12/H3
Sigla Settore Scientifico- Disciplinare: IUS/20

Table of Contents

Some parts of this work appear in:

Federici C. "Cloud Data Imager: a unified answer to remote acquisition of cloud storage areas", Digital Investigation Elsevier 2014 http://dx.doi.org/10.1016/j.diin.2014.02.002

Federici C. "AlmaNebula: a computer forensics framework for the Cloud", Procedia Computer Science Vol.19 pag. 139-146 Elsevier 2013 http://dx.doi.org/10.1016/j.procs.2013.06.023

1. THE RESEARCH PROJECT

1.1. Motivation

This PhD thesis discusses the impact of Cloud Computing infrastructures on Digital Forensics in the twofold role of target of investigations and as a helping hand to investigators. The Cloud offers a cheap and almost limitless computing power and storage space for data which can be leveraged to commit either new or old crimes and host related traces. Conversely, the Cloud can help forensic examiners to find clues better and earlier than traditional analysis applications, thanks to its dramatically improved evidence processing capabilities. In both cases, a new arsenal of software tools needs to be made available. The development of this novel weaponry and its technical and legal implications from the point of view of repeatability of technical assessments is discussed throughout the following pages and constitutes the unprecedented contribution of this work.

1.2. Introduction

Cloud Computing is a business model which advocates Information Technology as a service consumable on demand rather than as an endless pursuit to assets purchase. The idea of computational resources delivered proportionally to user needs and accordingly charged dates back to the mid sixties, but during the last decade only the remarkable advances in data center management have entailed economies of scale able to drop fares and level them to utilities such as water or gas. The Cloud is changing the way companies and public administrations are approaching IT and seems eligible to play a role so revolutionary to be compared to other milestones of technological evolution like the Internet or mobile telephony. One of the side effects of this overwhelming rise is a major impact on Forensic Computing, the science that deals with techniques and procedures to handle electronic equipment as a possible source of evidence in a trial. From one side, the pervasive availability of cheap cloud computing services for data storage, either as a persistence layer to applications or as a personal store for documents and pictures, is remarkably increasing the chance that cloud platforms potentially host evidence of criminal activity. When this happens, collecting data in a way that is able to resist to legal and technical vetting may reveal itself very tricky, because forensic tools targeted to cloud infrastructures are still in their infancy and issues concerning jurisdiction may apply. Relevant data may indeed be fragmented in countless shards, possibly available for a very limited timeframe and residing in more than one country. Furthermore, it is common practice for cloud providers to rely on services delivered by third parties (as in the case of Dropbox leveraging Amazon's Simple Storage Service): this may force investigators to potentially turn to more CSPs, possibly residing in different countries, in order to request registration forms, log files and ultimately raw data. Once presented a proper court order, cloud providers would be in the best position for extracting relevant data from their platforms in the most reliable and complete way. However, this kind of services are not so widespread to date and, therefore, the need to adopt a structured and forensically sound approach calls for an innovative software weaponry which allows remote acquisition of storage accounts by leveraging the low level programming interfaces exposed by providers. From another side, the Cloud may constitute a formidable ally to forensic investigators. Its massive computational power and storage capacity can be harnessed to achieve elastic scalability, fault tolerance and timely results from analysis activity, so to conveniently master huge amounts of digital evidences that otherwise could be impossible to wield. Indeed, traditional tools running in standalone or a client-server environment may fall short when handling the multi terabyte scale of a complex case or, conversely, lie mainly underutilized when dealing with few digital evidences. This matches the reduced willingness of budget constrained decision makers in investing capitals for building new datacenters and therefore boosts the appeal of business models like Cloud Computing that propose the concept of IT as a pay as you go. The Cloud rests on a solid foundation of well established technologies, but is a giant leap compared to classic hosting when it comes to availability and self service provisioning of resources. There is something really new under the sun. E-commerce platforms, large scale web site indexing and social networking have forced the pioneers of IT like Google, Amazon, Microsoft and Facebook to rethink the very meaning of managing a data center in order to tackle the "Big Data" issue:

- distributed file systems running on many networked commodity servers that allow to sum up the cheap directly-attached storage and efficiently compensate for failures even of an entire rack;
- NoSQL databases (Strauch, 2011) that waive to the strict ACID[i] compliance of relational DBMS, but in return achieve a gorgeous scalability over many nodes with impressive write performances and overall availability;
- parallel programming models that split the input data into chunks that can be processed concurrently by many computers and finally consolidate the results;
- pervasive scripting that allows achieving a high level of automation so that one single administrator can manage hundreds of machines or more.

To leverage all the benefits offered by ICT as a service, a new category of forensic distributed applications are needed though, where a variable amount of fairly affordable computers are opportunistically engaged to share a slice of the overall computational burden. The established calling convention of digital evidence will be useful in the following to address all the electronic devices that might be relevant in a criminal case and which elevate to the rank of proofs only when their evidential contribution is ascertained in a court of law before an unbiased judge.

1.3. Research objectives

Concerning the Cloud as a target for investigations, the research concentrates on studying the maturity level from a forensic standpoint of the programming interfaces published by providers to allow remote retrieval of content. It is interesting in particular to assess which capabilities are offered to find deleted files and past revisions of documents, protect access from accidental modifications and retrieve objects metadata, the "data about data" which may locate a user action in a specific point in time. This preliminary assessment will lead to devise the requirements and blueprint of a novel forensic tool which allows the examiner to navigate from a remote workstation inside personal cloud storages and make a faithful logical copy of content and metadata to a local mass memory in a way that could be called "*cloud dd*"[ii]. A prototype desktop application, namely *Cloud Data Imager*, has been developed which offers a read only access to files and metadata of Dropbox, Google Drive and Microsoft OneDrive storage facilities, allowing directory browsing, file content view and imaging of remote folder trees to local memory devices with export to widespread forensic formats. During this journey, there will also be room for revisiting the concept of repeatable technical assessment according to the Italian Code of Criminal Procedure (CCP), art. 360 CCP and 117 of Implementing Provisions of CCP, in the case of remote acquisitions of cloud data. Indeed, when dealing with physical mass memories there is always a chance of evidence damage, if a proper preservation and handling policy was not in place. The very action of powering an evidence could also damage it permanently because of electrical shocks. Conversely, these concerns do not apply for cloud infrastructures which are nearly always available and fault tolerant. Therefore it will interesting to evaluate if a remote acquisition repeated over time will lead to invariable results or evaluate the importance of occurred modifications. Coming to the role of Cloud platforms as helping hand to cleverly analyze a vast amount of digital evidences gathered from a case, the novel contribution of this work consists in discussing the design goals, technical requirements and architecture of *AlmaNebula*, a conceptual framework for the analysis of digital evidences built on top of a Cloud infrastructure and able to suit the needs of a small unit as well as a structured forensic department. This aims at embodying the concept of "Forensics as a service", a type of service offered by a cloud infrastructure where evidence devices are uploaded to provider's premises (most likely in a private or community deployment scenario) and their content is extracted, processed and made available to analysts by means of intuitive interfaces in order to allow detection of actionable knowledge.

1.4. Project outline

The rest of this work is organized as follows: next chapter deals with the Cloud Computing business model and discusses its foundations, benefits and risks. The third chapter gives a due background information about the national strategies devised by some countries to grasp all the relevant opportunities that it offers. It does not happen by chance if the nations who are best positioned to contribute to cloud forensics are generally the ones that already devised a formal and structured approach to evaluate cloud technology adoption. Chapter 4 opens the core discussion concerning forensic subjects as it delves into the main aspects of investigations targeted towards cloud storage areas, discussing the limitations of currently available tools and detailing the requirements applicable to a novel forensic software fit for remote acquisitions. The internals of Cloud Data Imager complete with on field tests are presented and a discussion concerning the comparison to the traditional seize/bit-copy approach is included. Chapter 5 deals with repeatability issues in the context of a remote acquisition scenario. Chapter 6 surveys the state of the art of free and open source forensic tools for evidence analysis, lists their limitations and presents *AlmaNebula's* core concepts, requirements and architecture. Conclusions are drawn in chapter 7.

2. CLOUD COMPUTING

People are routinely confronted with the alternative to make or buy something and usually the decision depends on the balance of key factors like quality, cost and delivery times. As an example, one could decide to reserve a deposit box in a bank instead of placing a safe in a wall of his house. This might happen because the need of storing valuable goods is limited in time or to avoid annoying masonry works. For the same reason, instead of equipping a fully functional but maybe normally undersubscribed data center, a company could decide to charter ICT assets like storage or bandwidth according to a profile that closely matches its needs: more power during demand peeks and partial or total release of resources when exigency declines. The possibility of consuming ICT as a tailored self service is one of the main features that distinguish Cloud Computing (CC) platforms from traditional forms of outsourcing, where a much tighter and less timely interaction with the provider was needed to size the necessary computing power. It appears that the credit to have pronounced the word Cloud Computing (CC) in its present meaning goes to Erich Schmidt, former Google's chief executive officer. In August 2006 at the Search Engine Strategies Conference he talked about an emerging business model where the computation and the data were hosted by servers located ".. in a cloud somewhere". Beyond the popular representation that look at the cloud like as an opaque container of data injected and retrieved by any internet enabled device, Cloud Computing strives to embody the concept of "ICT as a service": a constant availability of storage, computational resources and software platforms that are delivered to the final customer through a network, scale in and out dynamically and are charged only for the time of real utilization, resulting in no upfront cost or long term commitment. ICT resources that can be opportunistically engaged and decommissioned at user will with little or no intervention of the entity that owns or manages them, the cloud provider. In this respect, one could be misled into thinking that, once removed the topping made of a thick layer of marketing hype, what is left is the old seasoned outsourcing business model just rebranded. Indeed, not all cloud offers were made equal and the term *Cloudwashing* was just coined to address those services which do not comply to cloud platforms key characteristics, but however are advertised as such because none would otherwise consider them by now. For certain, it is very difficult having brand new ideas, especially in the ICT arena. The concept of a cheap, elastic and virtually infinite computing power that could be reached remotely dates back to the mid 1960s (Parkhill, 1966) and largely anticipates the formal definition of cloud computing set forth by the NIST in 2011 (see next paragraph). It is also indisputably true that CC rests on well established foundations as many of its building blocks have a long track record: economy of scale, resource sharing, disaster recovery or machine virtualization have been devised decades ago to tackle ever green problems that haunt the dreams of Chief Information Officers (CIO) all over the world: continuity of operations, low data center average utilization, intermittent and disrupting demand peaks, long delays in procurements of assets and, most importantly, the chance to be relieved from ICT management in order to take care of company's core business only. However, there is much more to it. New economic, infrastructural and technical drivers only in recent times have realized what fifty years ago could just be imagined and promise to make CC one of the most important business models in IT history:

- **The cloud pioneers**: thanks to the work of precursor companies like Amazon and

Google, the long held dream of the computer as a pay per use utility like electricity or water, has made true (Parkhill, 1966). Affordable services, ranging from plain disk space availability to fully fledged virtualized infrastructures, are now at reach of every user because of unprecedented improvements and optimizations of data centers management. Economies of scale due to massive purchase of goods, use of commodity material instead of expensive redundant equipment and tight cooperation between software development and system administration teams has significantly lowered the operative expenditure needed for running the infrastructure.

- **Budget constraints**: considerations above match the persistence of the financial slump that is hitting very hard during these years. Economic resources lack and decision makers, routinely struggling with shrinking budgets, have much less aptitude than in the past in investing capitals for buying or refurbishing some likely underutilized server farms. Under some circumstances it is better renting than facing the fixed costs of owning. This increases the interest for business models like cloud computing that promises to drop the time to market when starting new projects thanks to its flexibility and speed in provisioning and releasing computation resources.
- **Internet bandwidth**: generically available Internet access speed grew by something coarsely close to two orders of magnitude from 2000 to 2011[iii] and therefore is getting more and more feasible moving to the cloud applications that historically abode in corporate local networks only.
- **Cloud software development**: A virtually infinite computing power would be pointless if not properly backed up by an appropriate software offering. A large portfolio of ready-made applications and development platforms which are granted an

ubiquitous access, have dropped production delays and avoided many nuisances stemming from a per machine installation and update of packages.

2.1. The NIST definition of cloud computing

Even if there is no universal agreement on what CC exactly is, there is a widespread acceptance of the definition given by the US National Institute of Standards and Technology NIST (Mell and Grance, 2011), the organization identified in a key paper by former US CIO Vivek Kundra (Kundra, 2010)a as the authority appointed to guide US Public Administrations in their migration path to cloud services. According to NIST, Cloud Computing has five fundamental features, three service models and three deployment models as depicted in figure 2.1.

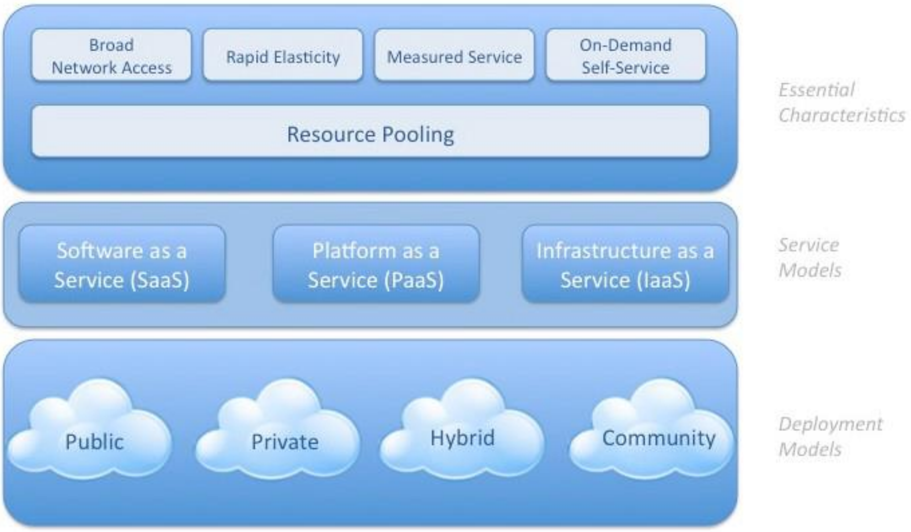


Figure 2-1 Features, service and deployment models of CC according to NIST

2.1.1 Essential characteristics

- **On demand self-service**: computing resources can be unleashed by the customer when needed and without human interaction with the cloud provider. This is a giant leap compared to the classic outsourcing model that usually required the reservation of ICT capabilities beforehand and entailed a strong dependency from the provider.

CC brings loose coupling in resource provisioning, although not complete independence. For instance, some provider enforce a baseline policy which limits the number of virtual machines that can be started concurrently (20 in the case of Amazon Elastic Compute Cloud EC2). Increasing this limit is possible, but can require a sort of out of band interaction with customers like filling a request form. We further have to notice the lack of *programmability* as a necessary feature of a cloud platform, that is the exposure of the infrastructure capabilities through library calls that could be invoked inside user programs. This allows an additional degree of freedom compared to prebuilt control panels already offered by the cloud provider. The authors of NIST special publication 800-145 possibly included programmability in the Self Service characteristic, but nevertheless we feel that this concept is so important to deserve an explicit citation as an autonomous feature.

- **Broad network access***:* cloud resources have to allow a network access (not necessarily the Internet) via protocols that facilitate the usage of the broadest spectrum of remote terminals. Usually cloud platforms are reachable through REST[iv] or SOAP[v] web services, encapsulated in HTTP[vi] or HTTPS[vii] payloads.
- **Resource pooling***: A* single instance of a resource is shared among many customers using a multi-tenant model that enforces isolation of customers' data. This means that if the resource is an application, the code base is the same, but appearance is personalized and data are segregated, usually using database tables. In case of hardware components, multi-tenancy is achieved through virtualization that inherently separates users at operative system level. As the word *Cloud* suggests, the geographical location of assets is transparent to the final users that only may decide to

confine them at macroscopic level, as in the case of data handling in a specific country for regulatory compliance. For instance, to meet their legal obligations, Amazon EC2 customers can place their resources, such as virtual machines, in one or more of ten *Regions* worldwide[viii]: Asia Pacific Tokyo, Asia Pacific Singapore, Asia Pacific Sydney, EU Ireland, South America San Paolo, US West Northern Virginia, US West Northern California, US West Oregon, US GovCloud and the brand new China Beijing. Regions are located in a single country and contain *Availability Zones*, which corresponds to data centers interconnected with high speed links. Users could decide to confine their resource in more than one availability zone of a region for disaster recovery purposes, but as the location of these zones is not advertized, a region remains the only landmark for resource placement.

- **Rapid elasticity***:* Cloud services can be elastically provisioned to quickly scale out in case of peak demand and be quickly released during idle times. This behavior allows one to arrange at need seemingly unlimited computing resources for a very reasonable amount of money and represents a great deal of efficiency compared to the static and often oversized data center that can be found in the average company. The property of "elasticity" is widely presented as a major breakthrough of this business model as one of the most generally appreciated characteristics of cloud computing resides in the fast provisioning of resources. Let's consider for example the Auto Scaling feature of Elastic Compute Cloud[ix], the computing platform of Amazon that enables customers to run concurrently up to thousands virtual machines (VM). Auto Scaling monitors a running instance's resources and is able, according to a predefined policy, to start automatically other virtual machines (within minutes) when, for example, CPU usage reaches 80% and, conversely, to stop them when load drops to 50%.

- **Measured service***:* Monitoring and reporting resource usage (e.g. CPU time, bandwidth or disk space) is a key point not only for auto scaling capabilities, but also to deliver a metered and transparent service to the final customer.

2.1.2. Service models

- **Software as a Service**: Customers rent readymade applications running in the cloud service provider (CSP) premises: office productivity, customer relationship management, sales, business intelligence and many more. Access can be granted from a variety of client devices or by means of a program interface. Users are offloaded from any management task and are thus able to focus on their core business, but conversely there is little or no capacity to influence key features like the format in which information is stored, with possible issues of data transfer back at a later time. Salesforce is just an example of SaaS.
- **Platform as a Service**: is the possibility for users to develop applications from scratch with high level programming languages like Java or C# that leverage CSP's hardware resources by mean of an Application Programming Interface exposed by the provider (usually proprietary). There are still no management tasks that are not related to application maintenance. Microsoft Azure and Google App Engine are an example of Paas.
- **Infrastructure as a Service**: It's a remote server farm made of virtual machines, pluggable block stores (external virtual volumes much similar to USB disks) and object stores (these will be extensively discussed in chapter 4 and 5 for their utmost importance from a digital forensics standpoint) at user's disposal as full administration rights are granted. Total freedom as to operative system selection and application

development of applications is balanced with an important burden of logical IT management shared with CSP that only retains control of the underlying infrastructure (from Virtual Machine Monitor[x] to physical security). Amazon Elastic Compute Cloud offers, among others, IaaS services.

2.1.3. Deployment models

- **Private cloud**: The cloud platform is run on or off premises for the exclusive needs of a single organization that can either be the owner and operator or can rely on third parties.
- **Community cloud**: Same as private, but here a group of organizations with shared interests and concerns come together to consume cloud services.
- **Public cloud**: The most common form. Cloud infrastructure is located at the premises of a commercial party which owns and operates it to sell its services to the general public. It must be considered that some cloud providers may rely on assets of others as in the case of Dropbox that rests on Amazon's Simple Storage Service.
- **Hybrid cloud**: Results from the composition of infrastructures belonging to the previous models. Each of them still stands as an autonomous entity, but is able to interoperate with the others by means of standard or proprietary protocols that allow migration of data and applications.

2.2. The true meaning of the cloud

As noted by Randy Bias, co-founder and CTO of Cloudscaling, elasticity is not a core propriety of the cloud, “but rather a side effect” (Bias, 2010). To bring cheap computing facilities to the masses at an adequate scale, companies like Google, Amazon, Yahoo and Microsoft needed to pioneer a new concept of ICT, bringing an unprecedented level of efficiency and cost effectiveness in running computation resources. In traditional datacenters, servers are confined in the same area for physical security and maintenance purposes only. They share air conditioning and power system, but have very dispersed hardware/software setups and management units that typically use commercial tools for every day operations. Each server communicates with a few others and can count on expensive high-end equipment such as enterprise class disks handled by array controllers with fault tolerance capabilities (RAID). The number of machines per system administrator is relatively low and changes due to new releases of applications are infrequent.

2.2.1. The warehouse-scale computer

Conversely, large Internet operators introduced the concept of "warehouse-scale computer" (Barroso and Holzle, 2009). This refers to clusters of hundreds of servers or more that run the same distributed application and behave as a single machine. Use of commodity hardware, such as 1 U[xi] servers equipped with directly-attached desktop class disks or ordinary 1 Gbit/s network switches, limits costs for provisioning, even if the inherently higher rate of failures of this material raises the problem of fault tolerance not at component level, but at server level. So redundancy is achieved by putting intelligence within the software and replicating data on many nodes that belong to separate clusters with a distributed file system such as Google's File System (Ghemawatt, Gobioff and Leung, 2003), so that the breakdown of a entire rack of machines would not affect service availability. This is in addition to the choice of keeping low the number of templates of hardware/software platforms not to make asset management a real nightmare. A unified administration team with a "DevOps"[xii] mentality which writes its own scripts completes the picture and allows reaching high level of automation, thousands of managed servers per system administrator and timely release of new application versions. The combined effect of automation, DevOps culture and use of commodity hardware brought elasticity as side effect (Bias, 2010).

2.2.2. DevOps

The DevOps culture (Edwards, 2010) harmonizes the activity of two historically separated corporate areas: development and operations. The former is in charge of creating new applications and has a mentality naturally open to change. The latter is requested to manage systems in order to create a safe environment for those applications to grant services availability. Changes to reliable setups are therefore perceived as dangerous because they can introduce bugs, security flaws and instability. The contraposition is increased when the two departments have separated office locations and when they report to unrelated managers. Developers usually work with rich graphical integrated development environments (IDE) that run in a single workstation or between few machines well connected with high speed local area networks. System administrators on the contrary work with server operative systems that may have poor user interfaces (possibly command line interfaces only), use scripting languages for every day maintenance tasks and deal with security appliances and slower wide area networks. Development process is targeted to functionalities and performance, less frequently to security. Consider the case of a new application for office productivity that is based on a communication protocol like Remote Procedure Call (RPC). In Windows servers RPC implementation features a dynamic port allocation according to which client and server agree to exchange data on a random chosen port whose number is greater than 1024.

When software is sent for production, operations guys will be probably disappointed because they do not know a priori which ports the application will use and will be forced to open an entire port range in the firewall, possibly exposing servers to attacks. So they will probably return the artifact to developers, stating that it is not suitable for production for security reasons and starting a tennis game that determines a complete waste of time. The mishap can be overcome with some workaround like "tunneling" (encapsulating RPC into single port protocols like HTPP that merely acts like a transport layer) or with some registry hack, but this is one of the uncountable examples of troubles that could be avoided with a tighter cooperation between the two teams. First of all, DevOps philosophy tries to disseminate a business culture among managers, software engineers, system administrators, testers and all other components of the production chain. All of them should be aware that they share a common goal of making high quality applications that fulfills the needs of customers. There is no room for working in isolation, no room for sentences like: "It works on our machines. Just lob the problem over the wall" (Nelson Smith, 2010). Theory is brought into practice by increasing contacts between software and deployment people (meetings, instant messaging, conference calls) and by adopting unified processes and tools, version-controlled software repositories and a lot of automation of lengthy and error prone manual tasks (Edwards, 2010) .

2.2.3. What does it take to build a Cloud?

Being a cloud provider means all of this. Before building a private cloud it must be carefully considered that the creation of a private or community cloud, overlooking the aforementioned key points of server farms organization and management typical of public deployments, means delivering undoubtedly useful scalable virtual machine services, but may be far from the effectiveness of a disrupting technology. One must wonder if it is feasible to uproot the current processes, unhinge consolidated, but maybe unproductive traditions, harmonize the work of development (often outsourced) and deployment (maybe outsourced to a different contractor or handled in house). Scale should not necessarily be viewed as a problem. One can think that it is needed a server population that is comparable to the one owned by a large web operator in order to raise a successful cloud infrastructure. This might not be true. The point here is not collecting millions of machines, but the way processes and people are organized. The critical mass can vary significantly and could not be so difficult to achieve, especially if more organizations come together. Just to start with a rule of thumb, one must consider the current number of server managed by a single system administrator in his organization (it should be a number ranging from 10 to 50 in the average) and multiply by a factor depending on the level of expected efficiency.

As Bias says:"*Are on-demand automated virtual machines an infrastructure cloud? I would argue no. That's not 'new'. Again, we need to look at what the large web businesses such as Amazon and Google did that has changed the game. It wasn't elasticity, it wasn't automation, and it wasn't virtual machines. It was a whole new way of providing and consuming information technology (IT). If you aren't following that path, you aren't building a cloud*" (Bias, 2010).

2.3. Benefits and opportunities

Why moving to the cloud? The decision is usually composite, as there are many problems of traditional IT that cloud computing business model is potentially able to address. Customers most commonly acknowledge benefits like increased computational capabilities, agility, reduced time to market and cost containment, even if not all of them could apply concurrently and it is necessary to pay attention to hype. Let's review the main benefits and stress the opportunities.

2.3.1 Lower barriers on entry

"Utility Computing" (Armbrust, et al., 2009) allows exploiting computing resources in the same way one can draw electricity from the power grid and pay for what it has been really consumed. For instance, Amazon Elastic Compute Cloud (EC2) charge its compute instances on an hourly basis and renting say a large virtual machine per 24 hours costs as much as renting 24 machines for one hour[xiii]. Of course you can reserve instances for one or three years and get a discount on hourly rates, but it's not necessary. There is no mandatory long term commitment, no upfront cost and it is therefore possible arranging a running IT infrastructure much earlier compared to traditional asset purchase. This possibility is very attracting for private companies, but for public bodies as well, especially for local administrations.

2.3.2 Elastic and reliable information system

It may seem weird, but according to VMWare in an average datacenter most servers withstand only a fraction of their maximum sustainable load, typically between 5 and 15%[xiv]. This is due to the fact that server population is usually shaped in order to withstand peak loads which can exceed the average from two to ten times (Armbrust, et al., 2009). Over-provisioning is the only way to avert system outages in case of unexpected workloads and lose immediate and potential revenues. With utility computing scaling in and out of theoretically unlimited resources in a matter of minutes is a provider's task, whereas customer is required to arrange some clever clauses in the Service Level Agreement and monitor their application as much as possible. Coming to reliability, it must be acknowledged that a cloud provider with a sound pedigree can offer continuity-of-operation capabilities that are much superior to the ones possibly deployed by most organizations. Best-in-class fault tolerant systems, backups and disaster recovery policies can make the customer achieve availability percentages of 99.9% and more.

2.3.3 No procurement hassle

Buying services instead of purchasing iron means turning fixed costs for procuring and maintaining IT equipment (such as power or personnel) to operative costs (Etro, 2011). This is known as translating CapEx (Capital Expenditures, which occur to acquire or improve an asset and can usually be deducted during some fiscal years) into OpEx (Operational Expenditure, such as license fees or bills, that are needed for running the infrastructure and can be deducted in the same year they incurred). Some studies show that, on average, 65% of annual IT Capex and Opex is necessary just for managing existing systems, draining financial resources that could be invested on new initiatives (Milne, 2010). Introducing new artifacts into the logistics cycle may result in an expensive and long journey for an organization, starting from market inquiry till assets disposal. When possible, avoiding procurements costs and delays can be a giant leap forward, especially for understaffed departments. Furthermore, shifting from CapEx to Opex may involve less troubles, because empowering the information system by purchasing new hardware usually needs a detailed planning to be presented in advance for approval, whereas expenditure needed for running the infrastructure are usually are taken for granted and authorized with less pain.

2.3.4 Delegating IT related workload

Notwithstanding the principle that the management of an organization is considered ultimately accountable for damages to people, assets and reputation resulting from security incidents, it holds true that many IT related workloads can be delegated with a reasonable level of peace of mind if: 1) an accurate risk assessment of all valuable assets, including a correct identification of the security class of data have been performed; 2) a cloud service provider has been selected that holds certifications by reliable independent authorities; 3) an agreement upon an accurate SLA with provisions for properly secured and resilient services has been reached; 4) service quality is monitored to the maximum extent possible, also appointing third parties to audit provider’s security controls.

2.3.5 Revamping old applications

Cloud migration can help recovering versatile IT professionals permanently staffed to run the infrastructure and relocate them to possibly meet their expectations of new assignments. This may strengthen research and development teams and offers the opportunity to revisit exhausted legacy software that shows well known limits, but that maybe no one dares to tweak because of poor documentation and fear of unpredictable results. Instead, a ground-up rethinking of an application in a modern environment using best of breed development tools can make it more responsive, available and tolerant to peak workloads. As an example of an old application that couldn't be adapted to keep the pace with today’s "must have" features like ubiquitous access and social networking integration, consider the case of the Army Experience Center (AEC), a pilot program created to improve the effectiveness of recruiting operations by leveraging the new technologies (Kundra, 2010)b. It was clear that it was impossible to upgrade the current Army Recruiting Information Support System (ARISS), an over ten years old proprietary platform, as the Army required a customer relationship management system (CRM) integrated with Facebook and that could be accessed by recruiters from different clients including mobile devices such as notebooks, smartphones and tablets. As a new platform delivered by a traditional IT vendor quoted in excess of one million dollars, the Army chose a customized version of a CRM tool by Salesforce.com, a SaaS cloud provider, fulfilling its needs at the cost of $54,000 per annum.

2.3.6 Creating business value

In its cloud computing strategy brief, Kurt Milne talks about "the IT Constraint Spiral" (Milne, 2010), meaning that IT is a natural target for funds reduction because business executives cannot often perceive the practical contribution of IT related activities to global company welfare. The return seems not proportional to spending as CIOs should exhibit better communications skills to explain IT capabilities to the management board. Fewer resources imply a diminished capacity of IT to accomplish its tasks and this unleashes a negative spiral that brings further shortage of funds. In order to break the spiral and make IT create business value, Milne says that IT must be not only cost effective, but also demonstrate that its spending is directly linked to company's revenues: IT must have a key role in boosting business critical applications. It's a three stage transformation roadmap, that includes as first and second step the virtualization of internal services (like file servers) and business critical applications to end with the creation of a private cloud where computational resources are fully virtualized and IT's contribute is strategic to create new products. Leaving behind the deployment model, that might not necessarily be a private cloud in all situations, we can assume that cloud computing is not only a mean for reducing costs and reusing resources, but it can help improve business applications, contributing to make them more robust, responsive and available.

The Cloud computing paradigm is perfect when considering applications that entail a demand of IT resources that is temporary or well localized during the month or time-varying with low averages or unpredictable (Armbrust, et al., 2009). Consider for example: 1) burst mode software like weekly reports generation or wage calculations that require a lot of CPU power for a limited amount of time; 2) temporary projects that last for a limited and well known timeframe; 3) promising projects whose validity is not predictable before having accomplished a test bed and that could either carried on or dropped; 4) new organizational units such as small departments that have the chance to "start small" with no capital expenditure and fairly reduced IT investment. Having on-premises application that run on virtual machines eases cloud adoption, no matter if private or public.

2.4. A still risky business

The path of an incautious migration to the cloud can be fraught with downsides, especially for complex organizations. Moving sensitive data to someone else's premises and out of our direct control recalls some understandable "ancestral fears", for example about privacy, integrity and availability of information. This sums up to the fact that technical safeguards aren't always as mature as they should be to grant a reasonable peace of mind. CC offering may be very heterogeneous, still poorly backed up by well-established standards as to portability of data or applications and can suffer from lack of transparency of some commercial subjects, especially when services are operated for the general public. Some issues are tricky: what is the applicable jurisdiction when data cross state boundaries? Is there a real way to measure the performance of the Service Level Agreement (SLA)? What happens if the provider exits the cloud business and customers are trapped in proprietary applications? A conscious approach to the cloud world is a very serious matter for an organization as it requires that decisions are taken in several fields in order to factor in business, technical, legal and security aspects. Legal issues particular constitute a severe hindrance at the moment as a real consensus on internationally shared norms concerning privacy and applicable jurisdiction is still missing and safe harbor agreements seem not enough to let sensitive data circulate freely. Ironically, some of the benefits of cloud that can represent a boon for an organization, such as disaster recovery capabilities, may potentially constitute its undoing. One example: replicating data across geographically dispersed regions increases chances of recovering from a catastrophic event, but can raise concerns on jurisdiction and compliancy to norms. Approaching the cloud is basically a risk management process, in which decision makers are requested to balance the costs of benefits and risks, managing the latter so that they can be well identified, understood and possibly reduced to an acceptable level. This implies, before signing a service contract, having in place a proper management framework to avert the occurrence of risks or at least mitigate their impact (Paquette, Jaeger and Wilson, 2010). As an example, in the following paragraph we will explore the case of the information systems of United States federal agencies, which needed a comprehensive risk management framework as a helping tool to minimize the issues stemming from the migration of services to the Cloud.

2.4.1. The NIST Risk Management Framework

Because of the nature of the information they handle, local and central government agencies must comply with precise norms to avert the risk of exposure, unauthorized access or unavailability of their precious estate. The United States Title III of e-Government Act of 2002 (Public Law 107-347), namely "Federal Information Security Management Act" (FISMA)[xv], provides an articulated framework that requires the adoption and enforcement of security controls about "*…information collected or maintained* *by or on behalf of the agency and information systems used or operated by an agency or by a contractor of an agency or other organization on behalf of an agency… ".* In this regard, the NIST was indicated as responsible for issuing standards and guidelines for federal information systems out of the competence of National Security. As a consequence, in January 2003 NIST launched the FISMA Implementation Project to produce guidance documents aimed at supporting federal agencies in:

- categorizing their information (by assigning a potential impact rate);
- providing an adequate protection level in accordance to its value;
- enforcing minimum security requirements in seventeen areas (such as access control, awareness and training, auditing and accountability, identification and authentication, media protection, incident response, personnel security and configuration management);
- performing an effective risk assessment and management.

NIST papers belonging to this project where published as Federal Information Processing Standard (FIPS) documents as well as Special Publication (SP) belonging to the 800 series. FIPS 199 in particular defines three level**s** of "*potential impact*s" on the interested organization in case of a security incident that should affect confidentiality, integrity and availability of data:

- **Low** impact, in case of limited adverse effect (that might cause for example minor financial loss or little damage to people).
- **Moderate** impact, in case of serious adverse effect (that might cause for example significant financial loss or significant harm to people, but without threats for the lives of individuals).

- **High** impact, in case of severe or catastrophic adverse effect (that might cause for example severe threat of mission fail or loss of human lives).

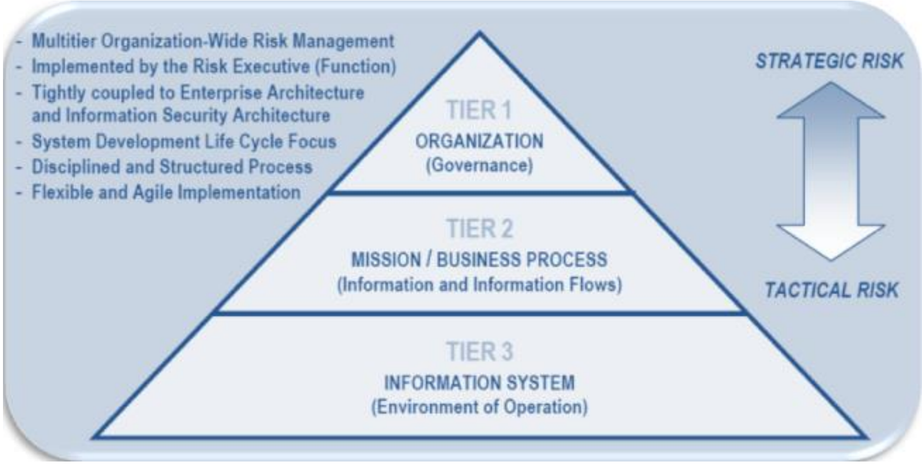
Impacts rating are preliminary for assigning a security category (SC) to different information types. Consider the fictitious case of a federal environmental protection agency whose SC relative to institutional documents might be:

SC = {(confidentiality, *LOW impact*), (integrity, MODERATE *impact*), (availability, MODERATE *impact*)}

When rating the overall security category of an information system, it is necessary to list all the SCs relative to all different information types and select the highest potential impact or High Water Mark (HWM) for every class: confidentiality, integrity and availability. In the fictitious case presented, if there are different kinds of stored data whose confidentiality maximum potential impact is LOW, so it will be in the overall SC. The same applies for integrity and availability. It is worth noticing that some commercial Cloud Service Providers (CSP) received a FIPS199 MODERATE[xvi] level accreditation and authorization from the General Service Administration.

A fundamental document on risk management was developed by the Joint Task Force Transformation Initiative Interagency Working Group and published as NIST Special Publication 800-37 Rev.1 (The National Institute of Standards and Technology, 2010). As management of risk is an organization-wide activity, the paper proposes a three tiered approach (Figure 2.2), in which every layer normally takes inputs from the previous (even if more complex dynamics could see discussions at a peer level) and, going from top to bottom, the area of interest continuously change from strategic to tactical:

- **Tier1** (***organization level***) deals with risk from an organizational point of view. Here an high-level risk management strategy is devised that includes the methods to assess all relevant security risk types (e.g. related to information systems, procurement, statutory compliance, legal, operations and reputation protection), the measures to mitigate identified threats, the definition of acceptable risk levels and monitoring;
- at **Tier 2** (***mission and business process level***) tasks are closely associated with Federal Enterprise Architecture (FEA) Reference Models (Office of Management and Budged, *"FEA Consolidated Reference Model Document",* rel. 2.3) and Segment and Solution Architectures (OMB, *"Federal Segment Architecture Methodology",* Jan 2009). Here a global information protection strategy is developed based on high-level security requirements, after having defined business processes, the priority of these processes according to organization's goals and the type of information needed to accomplish the task;
- at **Tier 3** (***information system level***) proper management, operational and technical security safeguards are deployed to all relevant information systems components according to NIST Special Publication 800-53 Rev. 3 *"Recommended Security Controls for Federal Information Systems and Organization".* Mostly at this level operates the Risk Management Framework (RMF, Figure 2.3), intended as *"a disciplined and structured process that integrates information security and risk management activities into the system development life cycle".*



- Multitier Organization-Wide Risk Management
- Implemented by the Risk Executive (Function)
- Tightly coupled to Enterprise Architecture and Information Security Architecture
- System Development Life Cycle Focus
- Disciplined and Structured Process
- Flexible and Agile Implementation
TIER 1
ORGANIZATION
(Governance)
TIER 2
MISSION / BUSINESS PROCESS
(Information and Information Flows)
TIER 3
INFORMATION SYSTEM
(Environment of Operation)
STRATEGIC RISK
TACTICAL RISK

Figure 2-2 NIST Tiered Risk Management Approach

It is important that risk related activities are performed from the beginning of the system development life cycle to avoid a more expensive late remediation and, in every case, before the information system is operative. The RMF is composed by six quasi-sequential stages depicted in figure 2.3 and involves many professional roles like the risk executive, authorizing officers, chief and senior information security officer and information security architects. The order of execution may be changed according to organization's needs and can be interrupted by local loops as in the case of unsatisfying security controls assessment at step 4 that requires changes in implementation and brings back to step 3 and so forth. However, the last phase before putting an information system into operation must invariably be the acceptance of risk by an authorizing official. The process is described as follows:

- **Step 1**: **Information system categorization**. According to the mission and goals of the organization, each subsystem of the information system is categorized according to FIPS199 in isolation or as an aggregate of items (like a pool of servers performing the same functions). This task involves all levels in the organizations, including senior levels like the CIO. The risk executive informs authorizing officers about organization's risk strategy, dealing with aspects such as: level of risk acceptable by the organization, identified threats, potential impacts on people and assets, protocols and tools used to evaluate the risk and proposed mitigation policy. At this point guidance is offered by NIST publications: FIPS 199; SP800-30, SP800-39, SP 800-59 and SP 800-60. Each subsystem is described and documented in the security plan by recording for example: location and environment, performed functions, security category of types of data stored and applicable norms, type and versions of operative system and applications, owner and entity that operates the subsystem. The main outcome of this phase is a detailed security categorization (*low-impact, medium-impact and high-impact*) of each piece of the information system in a way that is consistent with organization's risk management strategy and protection of its mission and business.

- **Step 2**: **Selection of security controls**. Based on the identified security category of an information system, security controls are selected according to a baseline pool (*low, medium and high*) as described in NIST SP 800-53 Rev3. Controls belong to 18 families (17 described in FIPS 200 as minimum security requirements plus Program Management) organized in 3 classes (Technical, Operational and Management). Controls families deal with: access control, auditing and accountability, identification and authentication, incident response and personnel security just to cite a few. Further controls can be chosen to tailor specific organization needs and everything is recorded in the security plan including the reasons for selection. When services are outsourced, description of how these are protected by the external entity is produced and assurance is obtained concerning an acceptable level of protection and risk management by the provider. Relevant documentation is provided by NIST publications: FIPS 199, FIPS 200, SP800-30 and SP800-53. At this stage, a monitoring strategy at a predefined frequency of the chosen security controls is agreed upon that includes an effective configuration management and control scheme. At the end of this step, proper security controls, monitoring program and authorizing officials are identified. The security plan is reviewed and approved.

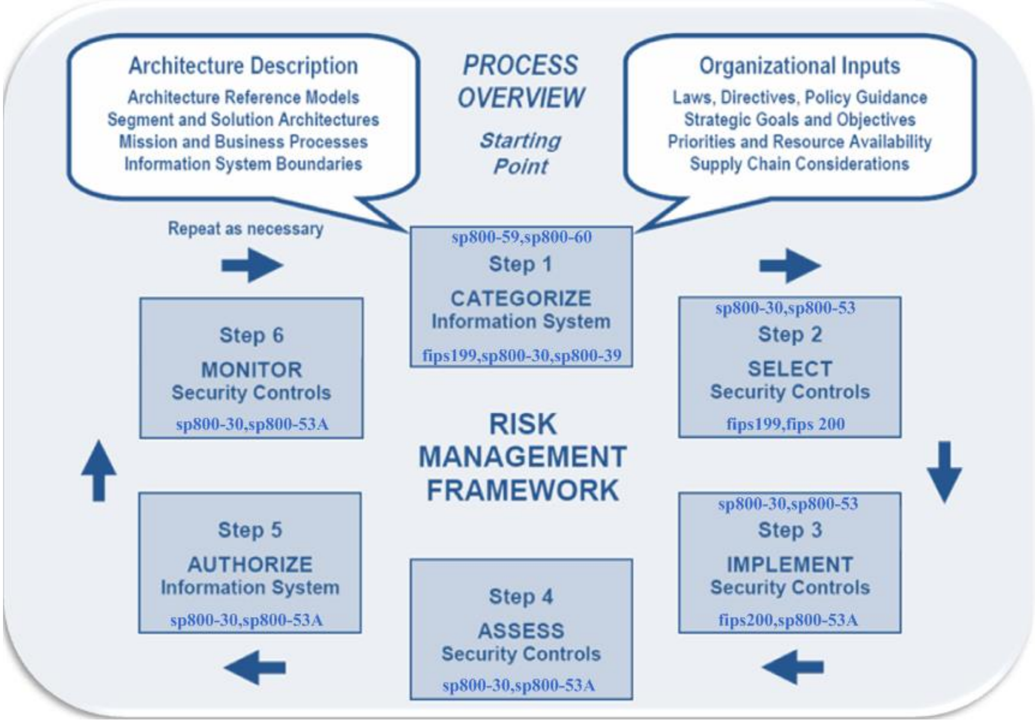


Figure 2-3 The NIST Risk Management Framework

- **Step 3**: **Implementation of security controls**. Protection controls are deployed to the proper subsystems and the security documentation is updated accordingly. Products should be used (e.g. antivirus or intrusion detection and prevention systems) that offer, if possible, a sound effectiveness pedigree after being approved by trusted third-party laboratories. When applicable, an information assurance activity is carried on to ascertain the quality of products such as design and development. Guidance: FIPS 200, SP800-30, SP800-53 and SP800-53A.
- **Step 4**: **Assessment of security controls**. An assessment plan with targets, procedures and tools is designed and approved. Assessing individuals or organizations are

appointed that offer the necessary level of skill, independence and confidentiality. Assessment operations are performed and reported. Remediation activities requested to address the most important weakness of controls are executed, toggling from step 3 to step 4 as necessary. Guidance: SP800-30 and SP800-53A.

- **Step 5**: **Authorization of the information system**. At this point authorizing officials can take their risk based decisions counting on strategic information delivered by risk executive at step 1 and having a *security authorization* package prepared by the information system owner, which contains the *security plan, the assessment report* and the *plan of action ad milestones.* This last document contains a detailed descriptions of the actions needed to remediate the flaws detected during assessment and a schedule with milestones within which problems will be solved. It is important to underline that, in case of security controls provided by external providers, authorizing officials' decision is based on information presented by the provider. The final *authorization decision document* indicates whether or not the information system got the permission to operate, along with terms, conditions and deadline of authorization. Guidance: SP800-30 and SP800-53A.
- **Step 6**: **Monitoring security controls**. As configuration changes are routinely applied to information systems, it is fundamental having in place a proper policy to track and document variations of hardware and software setups. Ongoing security control monitoring is needed to evaluate the potential impact of changes on security (for example the exposure to new threats or application of new controls) and keep authorization level as time goes on. Monitoring activity generates a report to the authorizing authorities that can be event driven or scheduled or both. As a

consequence, the assessment report is updated and so is the plan of action ad milestones when remediation is applied. Risk is therefore continuously evaluated and accepted. Guidance: SP800-30, SP800-53 and SP800-53A.

2.4.2. Risks and concerns

In the following, we are going to review the key factors to be considered in the evaluation of the overall risk of migrating to cloud services. Not each and every item could be applicable for every case, but the forthcoming list is comprehensive enough to cover most situations.

2.4.2.1 Loss of control

Perhaps the first unpleasant thinking which arises in the mind of cloud suspicious people is the perspective of being not the only master of data anymore. The circumstance of yielding a significant slice of sovereignty to some third party can be daunting. This is understandable because, even if a service level agreement is in place, doubts may arise concerning its efficacy and the possibility to monitor its real effects. Standard contracts are usually biased in favor of providers and laid down on a "take it or leave it" basis. A few customers will have the strength to contract clauses which states CSP accountability if sensitive data is exposed or lost. In this respect, the Public Sector can play an important role in fostering cloud technology adoption, as it happened for Internet wide acceptance and diffusion, thanks to its vast economic capacity (Allison and Capretz, 2011). Government organizations are in the position to achieve favorable conditions by setting forth contract vehicles and certification programs for providers that could be viable tools for the private sector too. For example, by massively purchasing email or office productivity services from SaaS providers that qualify after a remarkable path of scrutiny, the PS could be crucial to further lower prices, create case studies and convince private enterprises to do the same.

2.4.2.2 Value concentration

Cloud infrastructures are much like banks: they keep potentially extremely valuable information belonging to a large number of customers. This value concentration may make them a desirable target for any sort of cyber criminals as well as malicious insiders that could be tempted to break their vows of allegiance to their employer. As a part of their security policy and as permitted by law, well structured providers usually accomplish a scrutiny of the background of their employees which is proportional to their level of clearance. They also have in place an identity management system which audits all operations on customers' records and grants necessary privileges to relevant people only. After all, when a security incident occurs, a CSP may suffer no lesser damages than its users in terms of reputation and will strive to avoid security breaches. Nevertheless, any who owns some experience in running IT systems knows how difficult can be protecting from a bent system administrator, even if this holds true also when data reside in customer's house.

2.4.2.3 Physical security

From a cloud provider’s perspective, while costs for purchasing hardware and software can be balanced by the profits of serving many customers, the expenditure for physical security represents a rather important fixed cost that someone might overly wish to limit. When information systems move to the cloud, a detailed evaluation of the physical and environmental security of provider’s premises should accompany the assessment of IT security controls and business continuity policy. This is to ensure that equipment is physically protected, as the provider claims, by adequate measures like inconspicuous facilities, armed guards, armored rooms, video surveillance, alarms, biometric access control, visitors screening, power continuity, air cooling and fire protection. Despite this is situation is not new as it is typical of outsourcing, the foreseeable increase in cloud services adoption may exacerbate the problem.

2.4.2.4 IT Security

Cloud infrastructures are complex ecosystems which entail a huge degree of software layers possibly plagued by coding and configuration flaws which may pave the way to cyber attacks. As discussed in par. 2.1.1, resources are pooled among many customers and an isolation of domains is enforced that could be possibly bypassed by other malicious tenants in case of security vulnerabilities. Exposure to risks much depends from cloud service and deployment models: passing from SaaS to IaaS responsibility of securing platform gradually shifts from provider to customer. A possible threat model assumes that in a private/community deployment the computing environment is trusted and no harm can come from insiders and other tenants, till the time comes when a connection to a public network is operated. In a public cloud scenario this may change a lot, but this much depends on the chosen commercial partner: reliable operators can count on dedicated IT security teams, deploy detection points, analyze traffic for suspicious activity and prevents customers from probing other people's network connections. On the average, this may largely surpass the security measures that could be reasonably devised in a private corporate network and once again stresses the concept that migrating to cloud services needs to be a multi faceted informed decision.

2.4.2.5 Lock-in

One finds himself trapped in a cloud solution when it is not economically or technically feasible to switch to another supplier. This is a major point that many are happy to overlook until it's too late. Cloud offers still feature a large degree of proprietary solutions ranging from data formats, development libraries or interfaces and up to procedures. This is why it vital to consider in the first place the business impact, stemming for example from porting our flagship applications, should we decide to move away. The presence in the contract of some clauses concerning, among others, a painless export of data in a standard or well documented format is of paramount importance. When customers' records are stored in a format that comply with an international standard or at least at a well known or anyhow documented industry template, moving those data from one cloud provider to another may require much lesser effort. The same is true for applications, which can be run in both environments without the need to be coded again. Portability is achieved by removing dependencies on the underlying environment (Chetal, et al., 2011). In a well designed risk plan, portability issues must be faced since the beginning, as it is necessary to consider that the cloud provider one day could change and prepare the ground for a data transfer-back that is as smooth as possible. This means avoiding to be locked in proprietary material or at least include a proper clause to export data in a portable format. As it is cloud customers' common practice paying little or no attention to portability until they need to get their data back, Google[xvii] advices people to ask these three simple questions before using an application that will store their data: 1) *Can I get my data out in an open, interoperable, portable format? 2) How much is it going to cost to get my data out? 3) How much of my time is it going to take to get my data out?* The ideal answers should be: 1*) Yes. 2) Nothing more than I'm already paying. 3) As little as possible.*

Indeed, portability issues have always existed and that is why programmers adopt languages like Java that are operative system (OS) agnostic or adapt code to the underlying OS by mean of conditional compilation. Cloud models may exacerbate the problem however, because software platforms are not owned, but rather are delivered as a service whose cost/quality relationship could be unsatisfying over time. Lack of portability is a limiting factor for cloud diffusion not only in the case the organization decided to switch to another public cloud provider, but also if it wanted to move legacy applications to the cloud. Indeed, even if the software were written using a high level language like Java or C#, a large degree of code modifications may be necessary because cloud resources are possibly invoked by means programming interfaces that have many of proprietary extensions.

2.4.2.6 Troubles in ensuring an on premises-like protection

Leaving aside the case of on-premises private cloud, RMF's application can be challenging when cloud computing services come into play, because information systems are operated outside the security perimeter of the organization. This further degree of risk must therefore be addressed with additional security actions directed to ensure confidentiality, availability and integrity. For example, cloud providers can be contractually bound to implement all steps from 1 to 4 of the RMF (step 5 will mandatorily be an exclusive prerogative of the buying organization) and in the SLA all necessary security controls can be detailed. However, when the possibility to effectively assess those controls and monitor them continuously were practically denied, authorizing officials would indeed be forced to base their decision on papers and on trust of provider's reputation. This problem is real because verifying CSP actions like storage media sanitization before disposal or effective data confinement in the right geographical region could be very hard, if not impossible.

2.4.2.7 Data location and reachability

It is likely that data stored in a cloud provider facility cannot cross national boundaries, otherwise issues related to jurisdiction may arise. This is by law. So it is simply not feasible, from a legal point of view, replicating cloud data to other regions out of the borders and a proper SLA clause will state this in bold capital letters. All well then? Not quite. Even so, less tangible dynamics may cause data to leak out. Consider the case of a cloud provider headquartered in country A, but operating also in country B with a subsidiary company, that could be in the position to silently hand over data residing in datacenters of country B to law enforcement or intelligence agencies of country A. This situation is less theoretical than it could sound. For example, there is some concern in the EU, because U.S. headquartered cloud companies cannot possibly refuse to disclose privacy sensitive data physically located in European facilities to U.S. authorities under the USA Patriot Act[xviii]. Ensuring that under no circumstances cloud providers will make data reachable to foreign countries is of utmost importance.

2.4.2.8 Data sanitization

As discussed, one of the advantages of turning to an external provider is avoiding the burden ICT procurement and the associated costs for decommissioning storage devices. A fundamental task to protect valuable institutional data is performing an assessment of provider's policy and techniques to clear, purge and destroy working and backup media before disposal according to best practices (such as NIST Special Publication 800-88). If applicable and technically possible, using strong encryption algorithms would save a lot of worries concerning the effective enforcement of sanitization methods.

2.4.2.9 Need of specialized security personnel

Complying with norms and regulations, like the ones implemented in the NIST SP800-37 RMF, could be difficult for an organization willing to rely on external cloud services, especially for small departments. Several risk management and security functions are needed that cannot be easily collapsed on a few employees: risk executive, authorizing officer, chief and senior information security officer, enterprise architect, information security architect, just to recall some. External security professionals can be contracted for this purpose, but we think that the availability of contract templates, like the ones devised by some countries like USA or Great Britain for the public sector, would be very helpful to reduce risk management burden.

2.5. Further discussion

2.5.1. Which deployment model?

If properly implemented, public clouds have all the potential to store information as safely as private data centers, if not more in some cases. Nevertheless, it is worth keeping in mind that public clouds are managed by commercial parties that sell their computational resources to the general public and there are situations in which it is necessary to turn to private clouds or even abandon the will to migrate, maybe limiting to server consolidation activities only. Preliminary risks assessment may bring to discard a public cloud solution due to:

- provider incapability to guarantee an acceptable level of confidentiality, availability and integrity in relation to the nature of data. Despite nothing forbids a commercial provider to setup a facility with military-grade security, it might be impossible for a police force to entrust investigative records;

- the amount of information involved (Misra and Mondal, 2010). If an organization regularly deals with a huge quantity of data, like science or cartographic laboratories, pushing that data to the cloud within the due timeframe may imply relevant costs for bandwidth and storage;
- the quality of service required (Misra and Mondal, 2010): real time services require specifying very stringent SLA clauses that could not be possible or worthy to accept by a cloud provider. Quite some time will need to pass before seeing a Paas application for air traffic control delivered on a platform of a commercial party.

When a private cloud can be the only viable solution, it is better keeping in mind that behaving like a profitable provider implies all the costs of ownership for procuring, licensing and operating the data center. The need to enlarge the user base to keep an excellent information system's utilization can then address towards community computing. Different organizations may decide to come together and deploy a community cloud solution that could fit their common exigencies. In the case of the Public Sector, a central agency could deliver services to an entire department or to bodies belonging to the same business area. One example is the U.S. Defense Information Systems Agency (DISA) Computer Services Directorate (CSD) that supports the Department of Defense (DoD) in the Information Technology area. DISA created a private cloud infrastructure named Rapid Access Computing Environment (RACE)[xix], a IaaS platform that allows all Defense actors to purchase virtual machine services by mean of a self service storefront, in order to test and certificate software packages for acceptance to Defense Enterprise Computing

Center production environment. Hybrid cloud deployments could be another possible solution. Consider the case in which an organization owns a private cloud that is designed to handle the average load, but is under-provisioned to withstand occasional demand surges (*cloudbursting*). Peak traffic can be managed by launching as many as necessary instances located in the provider's premises and stopping them when exigency ends. If peaks occurrence is a relatively rare event, this setup guaranties performance and availability of applications at a little extra cost. Hybrid computing is also valuable in case of huge amount of data coming from different sources, which contains a lot of irrelevant noise that can be filtered greatly in house before being pushed to the public cloud for further analysis, consolidation and presentation to consumers.

2.5.2. Which service model?

The message is: "More control, more responsibilities", which means that the level of abstraction drops when moving from SaaS to IaaS and a greater degree of control and responsibilities is transferred from the provider (see fig.2.4).

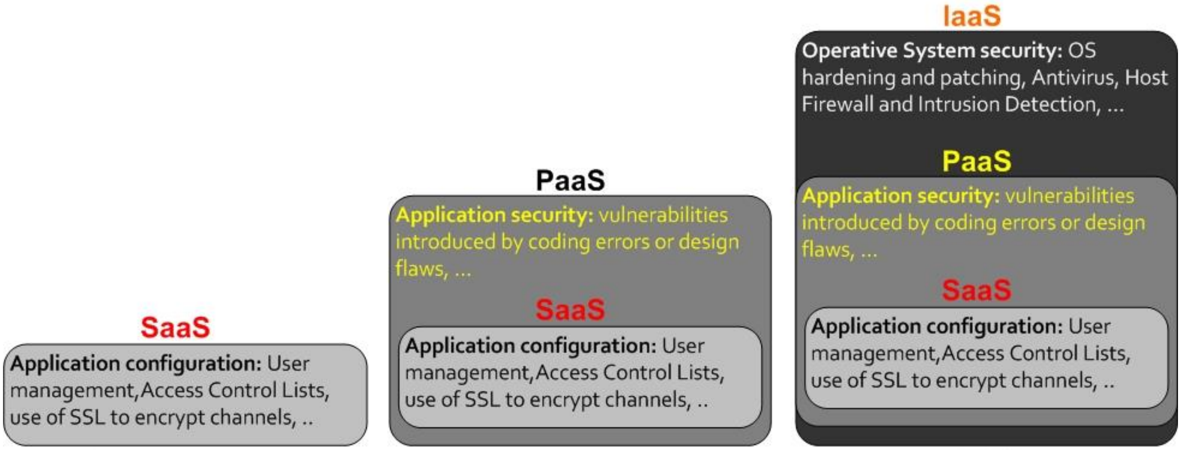

SaaS
Application configuration: User management,Access Control Lists, use of SSL to encrypt channels, ..
PaaS
Application security: vulnerabilities introduced by coding errors or design flaws, ...
SaaS
Application configuration: User management,Access Control Lists, use of SSL to encrypt channels, ..
IaaS
Operative System security: OS hardening and patching, Antivirus, Host Firewall and Intrusion Detection, ...
PaaS
Application security: vulnerabilities introduced by coding errors or design flaws, ...
SaaS
Application configuration: User management,Access Control Lists, use of SSL to encrypt channels, ..

Figure 2-4 User responsibility growth from Saas to Iaas

SaaS and PaaS are perfect as they allow an organization to focus on its core business, without caring about the low level plumbing of information system's maintenance. At

SaaS level users are relieved from any trouble concerning security or compliancy, but get prepackaged software that could cover only a part of their business needs and may have little or no capacity to influence fundamental aspects like data formats or protection techniques (at most they can ask for a customized version). With Paas customers are in charge to correct vulnerabilities introduced by design flaws or coding errors and configuration weaknesses caused by improper management of authentication or privileges (Mather, Kumaraswamy and Latif, 2009). Furthermore, the issues on application portability due to proprietary APIs must be considered. IaaS gives a large degree of flexibility, but provider takes care from the physical security of the infrastructure just up to the Virtual Machine Monitor[xx] (VMM). Customers have full administrative rights granted for the virtual machines they run and therefore are completely responsible for keeping safe and sound the entire software chain, from the guest OS to final user's application software. For organizations that have already started a server consolidation path, turning to a IaaS private or public solution can be almost straightforward as cloud providers usually give the possibility to import to their infrastructure customer's VM in the most widespread format. This way legacy software that could be difficult to upgrade or adapt to Paas environments might seamlessly run in the cloud, provided that all interdependence with other applications and services has been assessed in advance.

2.5.3. How is it possible to mitigate portability and interoperability issues?

Despite lack of portability is still viewed as limiting factor of cloud spread and there is much work ahead, it must be acknowledged that many efforts towards portability and interoperability were made in recent years. The following is by no means a comprehensive gallery of initiatives:

- the Open Virtualization Format (OVF) is an open standard, secure and portable format to package and distribute virtual machines (Distributed Management Task Force, 2010). It has been designed for platform neutrality, which means that virtual machine files in OVF format can be ported from one IaaS provider to another without modification. Furthermore, OVF has many important features as it allows the following and more: package efficiently a complex environment made of one or many interdependent VMs, verify integrity and authenticity of the package by mean of digests and digital signatures, specify all details of the virtual hardware (like CPU or memory) and host relevant metadata (like virtual disk information, logical networks or license agreement for the software in the package).
- OpenStack is a community project started by Rackspace and NASA to build massively scalable IaaS clouds that could be viable for building private or community cloud solutions. OpenStack addresses cloud infrastructures interoperability by leveraging open source and open standard. It consists of three subprojects named: 1) **Nova**, an hypervisor-agnostic fabric controller to run and manage virtual machines networks; 2) **Swift**, a distributed store for objects like virtual machine images or pictures that can scale to petabytes (one petabyte is around one million of gigabytes) and achieves fault tolerance by replicating data across multiple cluster nodes made of commodity hardware. Unlike file systems items that can have byte granularity, Swift objects are atomic in nature, meaning that to be updated, an object needs to be deleted and uploaded again (more on this in chapter 4); 3) **Glance**, a service for register, discover and deliver virtual machine images via standard interfaces. The whole infrastructure is orchestrated by mean of authenticated web services which expose control of

hypervisor, storage and networking. The application interfaces are compatible with tools already used by commercial vendors and this paves the way to portability or hybrid cloud solutions, when applicable. More projects were added over time, for instance to manage network addresses and routing configurations or block storage which cloud be attached as volumes to virtual machines.

- In April 2011 the Institute of Electrical and Electronic Engineers have created two working groups named IEEE P2301[xxi] and 2302[xxii]. The former will develop a Guide for Cloud Portability and Interoperability Profiles (CPIP) to support cloud vendors, providers and developers to converge towards standardized application interfaces and file formats organized in groups of capabilities called *profiles.* The latter will focus on standards for Intercloud Interoperability and Federation (SIIF) to allow different cloud infrastructures to federate and interoperate.

3. CLOUD COMPUTING NATIONAL STRATEGIES

The foreseeable advantages of the cloud revolution are so promising that national or super-national migration plans are sprouting all over the world. This was also due to the consideration that some agencies in the public sector were already using cloud technologies as a "de facto" service, possibly threatening the overall information assurance level. So it was much more fruitful taking note of the situation and organize a coordinated, thoughtful and risk conscious approach that could give guidance and support to all stakeholders. During the whole preparation period, the cooperation amongst Governments, specialized public agencies, industry, academia and professionals was fostered in order to deliver an evaluation framework not only to policy and decision makers of Member States and public administrations, but also to small & medium business enterprises (SMEs) CIOs. These comprehensive policy documents, enriched with fictitious use cases and complete with ancillary technical specifications, usually cover the following areas: a clear statement of needs and requirements according to organization's mission, the jurisdictional and legal context with related implications and limitations, guidelines for a rigorous risk assessment, identification of applications eligible for migration and hints for the choice of a service (SaaS, PaaS, IaaS) or deployment model (private, community or public). In the following, we will deal with strategies of the European Commission, United Kingdom, United States and Australia.

3.1. European Commission

A lot of work has been done in the recent years to prepare the ground with solid recommendations for a wise adoption of a cloud computing strategy in EU member states (MS). The European Network and Information Security Agency released a report (The

European Network and Information Security Agency, 2011) aimed at helping government organizations to better scrutinize their needs (with particular attention to information security) and choose the best cloud deployment platform accordingly (public, private, community or hybrid). Practical use cases are discussed, with local administration and central government scenarios. The paper acknowledges the benefits stemming from public cloud providers in term of cost-effectiveness and resiliency, but regulatory issues advice limiting to non sensitive data and not critical applications. In this latter case, private or community infrastructure should be selected, if the requirement of adequate scale is fulfilled. Conclusions are in favor of a staged approach to cloud computing services as they match most of the requirements of public administrations, provided that managers at all levels undertake a thorough process for assessing the impact of all possible risks to processes and applications: loss of control, lack of compliance to laws and regulations and poor network connectivity in some residual areas of Europe, just to recall a few. An evaluation of strengths, weaknesses, opportunities and threats, known as SWOT analysis (Böhm, 2013), must be considered as bare minimum and must be completed with a security assessment as detailed in a previous paper (The European Network and Information Security Agency, 2009). At the World Economic Forum held in Davos in January 2011[xxiii] Neelie Kroes, Vice-President of the European Commission and responsible for the Digital Agenda[xxiv], while acknowledging the role of cloud computing as service model potentially able to change the very meaning of making business in the modern enterprise, announced an EU-wide strategy based on the following pillars:

- **Legal framework**: revision of EU data protection directive[xxv] , in particular concerning:
  - "the right to be forgotten", which entails new and revisited prescriptions to grant an

individual an effective right to withdraw consent to its personal data storage;

  - "greater transparency", about the nature and purpose of collected data;
  - "privacy by default", according to which protection remains even if the reason of processing data changes;
  - "data location invariance", so that EU laws apply to services consumed from EU member states territories, disregarding the geographical location of the data processor.

- **Technical and commercial fundamentals**: strong focus on information security, standardization of software interfaces and data formats, design of sample contracts and SLA between parties.
- **Market**: the Commission will partner with Member States (MS) to support innovative projects targeted to cloud platforms.

In May 2011, the European Commission launched a public consultation on cloud computing looking for opinions from companies, public administrations, academics and individuals concerning users experience, visions, opportunities and threats. The main outcomes of the consultation[xxvi] showed a large degree of legal uncertainty in cross border data transfers where it is not so clear which kind of jurisdiction applies. Consequently, the acknowledged need of more information on rights and duties welcomes any decision support tool like guidelines, checklists and standard clauses for service level and end user agreements. Provided that these clauses are simple to understand and clear in their wording, they could be usefully integrated in the final contract, especially within the EU. Surprisingly, respondents were largely divided about the trumpeted non homogeneous implementation at national level of the EU privacy directive which, according to the providers opinion, severely hampers a widespread cloud adoption. Furthermore, it turns out a general acceptance of the global nature of the cloud computing that would require a broad discussion concerning information transfer and treatment at the highest levels such as the G20. Finally, the need for a stronger research effort is widely recognized, especially in the area of time critical applications or hybrid cloud platforms management. Relying on this survey, in September 2012 the Commission issued a formal strategy document, namely 'Unleashing the Potential of Cloud Computing in Europe'[xxvii] , whose target is facilitating the adoption of CC in all sector of EU economy to cause by 2020 an overall impact on European gross domestic product (GDP) of about 1000 billion Euros and a consequential creation of nearly 4 million of new jobs.

3.2. UK G-Cloud

Definitely inspired by the U.S. "cloud first policy" (to be discussed next), in October 2011 the UK Cabinet Office released a document named "Government Cloud Strategy"[xxviii] (G-Cloud) as part of a wider ICT reorganization program[xxix] , aimed at delivering better services for the public at a lesser cost. Accomplishment of this target is feasible thanks to a greater efficiency obtained with the creation of a common standard based ICT infrastructure that allows resource sharing and reuse. The scope of this pioneering project is vast and ambitious as it is directed to save nearly 1.4 billion pounds during a four years period. This would be possible mainly by creating a more competitive, transparent and oligopolies-rid marketplace where all players will have the same opportunities to access procurement tenders, by reviewing of the most expensive ICT projects and by posing increased reliance on the expertise of internal workforce at the expense of external consultants. The G-Cloud program rests on the smart consideration that public cloud services, based on the concepts of scalability and pay per use, can be a flexible and cost effective first choice for many organizations of the PS, which often do not need costly bespoke ICT solutions when low cost mass market products can be leveraged with little or no customizations. G-Cloud services can be consumed from a private Government infrastructure as well as from trusted commercial providers that offer proper warranties in terms of information assurance. Hybrid scenarios are admissible too. Central to the realization of this strategy is the CloudStore[xxx], a online storefront open for business as of 2012, where public bodies can procure storage, applications, infrastructures and professional services by querying a catalog that reports service description, features, associated costs and business impact levels (BIL). Services implementation can vary greatly. For instance, a commodity hardware private cloud can be installed by the CSP on the customer's premises while retaining an "as a service" billing or deployed within CSP data centers located in the UK. When the sensitivity of data forbids turning to any commercial party whatsoever, the strategy states that the government may rely on its data center estate that, after a rationalization and consolidation phase, should remarkably improve its average utilization thanks to an increased asset sharing. Anyway, it holds firm the principle that each public body is ultimately responsible for the risks stemming from moving or creating its data in the cloud. Assigning a security category to the data it owns and choosing a cloud service with an appropriate impact level is a prerogative of each information owner, who must purchase a service suitable to information assurance needs. That is why a pan government CSP accreditation service[xxxi] carried out by the UK CESG[xxxii] (Communications-Electronics Security Group, a Government division that deals with information assurance) has been devised as a support tool with the goal of approving once to reuse the same service many times across the PS. For completeness sake, it has to be reported that the G-Cloud program faced some alleged problems at its inception that seemed to cause its cancellation. However, it is fully operational now (April 2014) and must be considered an ongoing effort that will probably converge to full maturity only iteratively. Indeed, it stands as a bright and tangible example of grasping the opportunities of innovation at the largest scale and particularly laudable is the commitment to transparency as suppliers are requested to provide reports of all invoices for procured services which are published every month[xxxiii]. Analysis of these raw data[xxxiv] shows that from 2012 to the end of February 2014 a total amount of 124 million pounds have been spent in 7558 transactions for procuring Infrastructures (Lot 1), Platforms (Lot 2), Software (Lot 3) or Professional Services (Lot 4). In this respect, it has to be noted that some spending is zero and some is negative possibly because they stem from compensations operated with suppliers for amounts already paid by some Administration. Therefore, if we anyhow consider all expenditures positive to evaluate the total value of cloud services we reach an amount close to 128 million. Compared to large companies, SMEs have been awarded of 59% of total sales by value and 58% by volume. The Central Government accounts for 78% of total sales by value operated by the PS. Table 3.1 reports the value of cloud services from 2012 up to the end of February 2014:

	2012	**2013**	**2014 (up to Feb)**
Total value	£ 7,125,014.52	£ 89,801,190.69	£ 30,787,950.53
Average spend	£12,521.99	£17,379.75	£16,897.89
Num of transact.	569	5,167	1,822
Min spend	£0.00	£0.00	£0.00
Max spend	£470,000.00	£800,000.00	£437,025.00
Std Deviation	£27,758.79	£40,599.83	£41,300.76

Table 3-1 G-Cloud services statistics from 2012 to 2014

Table 3.1 shows that compared to 2012, very first year of business of the CloudStore, during 2013 there has seen a vertical rise of the demand of cloud services both in value (nearly 90 million pounds compared to 7) and in volume (5,167 transactions compared to 569). The upward trend seems confirmed because just in the first two months of 2014 about one third of the value and volume of 2013 has already been produced. Partitioning the total value of transactions into lots as shown in table 3.2, it can been seen that there is a preponderance of professional services which accounted for more than 79% in 2013 and exceeded 80% in 2014. Software weighted for a gratifying 15% in 2013, which reduced to 10% in 2014 for the benefit of Infrastructures which doubled their importance, passing from 4.44% to 8.46% in 2014. Platform as a service turnover is almost negligible (less than 1%).

	2012	**2013**	**2014 (up to Feb)**
	Value		
Lot 1	3.74%	4.44%	8.46%
Lot 2	2.96%	0.91%	0.85%
Lot 3	30.39%	15.26%	9.98%
Lot 4	62.91%	79.39%	80.70%
	Volume		
Lot 1	104	661	332
Lot 2	53	101	23
Lot 3	140	848	267
Lot 4	272	3,557	1,200

Table 3-2 G-Cloud services partitioning into lots

To determine whether this trend will possibly continue during 2014 or if there is meaningful probability that this distribution of purchases into lots will be different, we applied a chi-square goodness of fit test to a randomly selected pool of 200 purchases performed during 2014 (observed counts in table 3.3) with a degree of freedom equal to 3 (the number of lots minus one). The null hypothesis will be that there is no difference among distribution into lots compared to 2013 as offsets between observed and expected values are only due to chance. The expected frequency is the number of purchases for each lot divided by the total volume produced during 2013 (e.g. for IaaS 661/5.167 corresponds to 12.79 %). Table 3.3 shows test findings:

Lot	Exp. freq	Norm.Exp.Count	Obs. count	(O-E)^2/E	
1	12.79%	25.59	40	8.12	
2	1.95%	3.91	6	1.12	
3	16.41%	32.82	27	1.03	
4	68.84%	137.68	127	0.83	
Total	100.00%	200	200	11.10	χ^2

Table 3-3 Chi-square goodness of fit test results

Choosing a confidence level of 0.05 or less, it can be calculated that a chi-square value of 11.10 corresponds to a p-value of 0.0112 which makes us reject the null hypothesis and conclude that it is reasonable to expect that distribution of lots will be different during 2014.

3.3. United States

In a pivotal paper (Kundra, 2010)a, Federal CIO Vivek Kundra listed the top 25 priorities to reform Federal IT. Drivers for change are solid:

- projects that exceed budgets and fail to achieve the expected results with detrimental consequences for the service due to the public;
- a galaxy of proprietary, locally managed systems that need a considerable amount of time and money to be provisioned and are largely underutilized (by less of 30% of server capacity).

Point 3 in particular, namely "cloud first policy" (CFP), has the purpose of engaging agencies CIOs in selecting three "must move" applications to be ported to public or private cloud platforms as appropriate. The first migration needed to be accomplished within 12 month and the remaining two within further 6 months, after having devised a detailed risk analysis plan. Meanwhile, a federal cloud computing strategy was put on paper (Kundra, 2011). The document reported that an estimated amount of 20 billion dollars, corresponding to the 25% of the overall Federal Government IT expenditure, could be potentially converted into CC services: this would result in 30% savings only for data center infrastructure that could be usefully repurposed. This well justified an articulated guide to support public administrators in grasping the benefits of the cloud: better asset utilization, lower maintenance costs, improved capacity to withstand IT resources demand peaks and increased agility in starting new programs with quick evaluation of projects feasibility (the "start small" approach). Since migration required a significant mentality change, shifting the view from assets to services, from risk adverse culture to an entrepreneurial approach, Kundra's vision delivered also a decision framework based on three pillars: "*select, provision, manage*". First movers needed to be services that concurrently exhibit a high degree of "value" and "readiness". A high value service, if moved to the cloud, allows achieving the best improvement of at least one among agility, efficiency and innovation. Underutilized applications, whose maintenance is difficult and costly, are an example of valuable assets in this respect. The term readiness is composite. An application can be deemed ready for migration when agencies have verified, according to their missions and compatibly with their capabilities to contract successfully, that one or more of the following requirements have been fulfilled:

- public or government provider's trustfulness in terms of compliancy to laws and agency's information processing standards adopted under the provisions of FISMA, event auditing and vulnerability assessments, confidentiality and integrity. To facilitate agencies decisions, the GSA (General Service Administration) initially identified reliable service providers that, after a comprehensive assessment and authorization program in accordance with NIST Special Publications 800-37 and 800-53 (rev 3), met all security requirements at FISMA's Moderate Impact Data security level. These operators were granted an Authority to Operate (ATO) on IaaS Blanket Purchase Agreement (BPA) and could sell storage, virtual machines and web hosting through the cio.gov[xxxv] portal. This accreditation process has been further finalized in the Federal Risk and Authorization Management Program (FedRAMP)[xxxvi], whose target is providing a standard and reusable approach to security assessment, authorization and monitoring for government and commercial cloud computing services. To achieve a provisional ATO issued by the Joint Authorization Board (JAB), an authority whose members are the CIOs of Department of Homeland Security (DHS), GSA and Department of Defence (DoD), a cloud provider must initiate from scratch a security assessment process compliant with NIST Special Publications 800-37 and 800-53 (rev 3). This process aims at selecting the appropriate physical/logical security controls (such as a biometric access control system or a firewall) and finally implementing and assessing these controls. All this activity is documented in an *security package* containing, as a minimum, a security plan and an assessment report. The CSP then contacts a government accredited "Third Party Organization" (3PAO) to independently verify the package. If all goes well, the package is submitted to the JAB and, if a provisional authorization is granted, it is inserted in a shared repository so that each agency can use it as a baseline to issue its own Authorization to Operate (ATO), adding additional controls, if necessary. This authorization path can be also started by an agency if the security package was not previously present in the repository. This “do once, use many times” behavior saves time and money because agencies will likely rely on a baseline of already granted government-wide authorizations and take care of their specific requirements only, if any;

- provider’s continuity of Operations (COOP) capacity. This could be endangered by a plethora of reasons including provider’s end of business, natural disasters or man crafted attacks;
- maturity of the provider’s offering and adherence to standards, to minimize lock-in issues;
- level of obsolescence: applications that needs to be revamped with new functions or that suffer in performances because of exhausted hardware are to receive a higher priority than others that were recently revised;
- suitability of legacy assets to migration: a well documented software that has little and clear interactions with other applications is a better candidate than one that accumulated layers of cryptic changes over the years and could exhibit an unpredictable behavior once removed from its usual environment.

The last part of the decision framework section reported recommendations concerning:

- an effective capacity of agencies to sign successful contracts with providers by putting on paper a Service Level Agreements (SLA) granting a prompt, secure and resilient service, with a specific clause that enables an independent third-party assessment of provider’s security controls. Contracts needed to be monitored for SLA compliancy and providers held accountable for service underperformance or disruption. SLA is an output metric that forces a mindset shift from assets to services.

- re-evaluation of the Quality of Service (QoS) delivered by the providers whose motivation needs to be kept alive by mean of periodical competitive bids. This requires that CIOs and their staff be always up to date about market developments.

As a final remark, it is worth noticing that the large body of directives of the American strategy points in the right direction to implement a quantum shifts in the ICT-mediated relationship between the American public sector and taxpayers. We did not expect anything less in the land of the pioneers of the cloud. The revolution is underway and appears irreversible, even if a huge cultural and organizational effort will still be necessary. Managers at all levels will need to be motivated and encouraged to further develop their expertise and entrepreneurial culture. It is known that all over the world some areas of the public sector are reluctant to changes as they don't want to endanger their stable position facing the associated risks. Here a degree of external guidance and control is necessary, but it seems not enough to achieve the desired results. What is needed is also a substantial endogenous spur, a drive stemming from the awareness of being part of a much larger project than the individual as the final prize is a greener, more agile, more reliable and citizen friendly public administration. According to Allen et al. (Allen, et al., 2004) the use of IT as a strategic instrument to better serve the public, instead of being a trivial tool to accomplish every day office tasks, depends on the organization of government agencies. They showed that a traditional rigid hierarchical structure inhibits the necessary horizontal collaboration and information sharing among departments.

There are some hurdles to overtake. For example, one of the consequences of purchasing computing power as a utility is that public managers will have to show a great capacity to negotiate profitable SLAs with contractors as now a complex service is delivered whereas once there was only internally managed bare metal. As previously discussed, standard contracts are usually biased in favor of providers. In what extent they will accept to be held accountable if sensitive data is exposed or lost needs to be agreed upon. The presence of some clauses could be vital to enhance security and avoid as much as possible the risks of lock-in: encryption methods for data in transit and at rest, free takeout of information in a standard or well documented format when contract expires and a documented list of possible third parties involved in service delivery. Anyway, U.S. public sector has the critical mass to achieve favorable conditions and, once more, pave the way for many other public bodies all over the world.

3.4. Australia

The demand of public bodies for flexible, cost-effective and performing services is increasing in Australia, a nation that poses much reliance on ICT and records an annual Government's expense of 4.3 billion dollars. The cloud computing strategic direction paper[xxxvii] aimed at giving guidance in evaluating opportunities and risks, but showed a more neutral view of the cloud than other countries, when it states that" *…cloud computing is just one of many sourcing models agencies should consider and is not necessarily a suitable replacement for all of their current sourcing models…*". Curiously then, the fact that the paragraph dealing with risks and issues came before the ones covering benefits and opportunities, seemed to warrant for a very meditated approach to this business model. The document ultimately underlined the opportunities lying in cost savings and increased agility, scalability and efficiency, further stimulated by the spread of the National Broadband Network (NBN)[xxxviii], but acknowledged as well that some aspects of CC such as contracts, regulatory compliance and security were still immature. The policy stated that agencies were allowed to shift services and applications in the cloud provided that they previously demonstrated that the game is worth the candle. This means not only ensuring that an adequate value for money is present, but also that the service is properly secured in accordance to the Australian Protective Security Policy Framework (PSPF)[xxxix], the Australian Government Information Security Manual (ISM)[xl] and the Privacy Act of 1988. Cloud services approach were a three phases process:

- phase 1: from 2011, an "enabling" preparations phase, in which agencies received guidance about policy, principles, risk-management and contracts with a lot of knowledge sharing. The output was a Provider Certification Program as it happens in the U.S.A.;
- phase 2: in parallel with phase 1, this stage involved public cloud service adoption. Australian Government Information Management Office (AGIMO) public web sites shifted towards the public cloud and this was the pilot for a Government wide migration strategy. Agencies were then encouraged to evaluate public/hybrid cloud offer and migrate application dealing with non sensitive data, when appropriate;
- The last strategic phase (mid 2011 onwards) encompassed Data Centre Strategy integration, a Government storefront (similar to U.S. cio.gov) and investigation/adoption of private or community clouds.

The Australian cloud strategy was not as articulated as the U.S.'s, but we surmise that the Government was looking at the cloud business with "wide open" eyes, taking the necessary time to ensure a meditated cloud adoption and exercising a strong degree of governance over agencies.

4. REMOTE ACQUISITION OF CLOUD STORAGE AREAS

4.1 Open Issues

A key role in the widespread diffusion of the Cloud has been played by distributed file systems and object stores, which allowed to reach virtually infinite storage capacity by summing the individual contributes of the disks placed inside commodity servers. Well known solutions exist, either proprietary or open source, that ensure high availability and geographic distributions of data. A side effect of a reliable and cheap storage area is the remarkably increasing chance that it can be used for harboring crime related data, such as credit card numbers, stolen identities or violated credentials. Unfortunately for the digital investigator, distributed architectures may entail difficulties when it comes to rebuild a global picture as files get partitioned in several chunks of configurable size and are scattered among a potentially vast population of participating nodes (Quick and Choo, 2013)a. This most probably prevents forensic teams from dirtying their hands with write blockers and bit stream copiers because it is hard to detect which of the plethora of nodes hold relevant data without digging into file system internals. But this is regrettably just a part of the story: proprietary technologies, unavailability of the provider to deliver a console with root privileges to third parties or simply lack of jurisdiction help figure out why an on-field approach may simply be totally unfeasible. So the natural conclusion should be serving a warrant to cloud providers as, in principle, they are in the best position to extract relevant data from their platforms. While this approach seems straightforward and rid of troubles, relying on a party that does not natively offer a professional forensic service, requires that a good deal of trust be placed on procedures and tools used at the provider's premises (Dykstra and Sherman, 2012). Data should be delivered to forensic investigators in a well known format, as complete as possible, integrity protected and non repudiable. Consider however the following scenarios where data acquired as a result of a warrant could be deemed unacceptable before a court for lack of reliability or sufficiency:

- a system administrator without a specific forensic background uses an ordinary maintenance script to restore the requested data from a backup. As a result, content gets extracted, but some file metadata are overwritten;
- deleted files are not recovered, even if this was technically possible;
- once packaged, the blob gets delivered without integrity protection codes or it is impossible to uniquely associate it to the provider because of flaws in the chain of custody;
- in case of proprietary templates, raw data is not exported in a well known format and browsing is only possible by means of a viewer program;

Resorting to the scrutiny of a third party appointed as needed to audit and certify the operation would result into additional costs and possibly further delays. Agreeing beforehand on an acceptable strategy for acquisition of data between law enforcement (LE) and provider could translate into delays as well and might need to be redesigned when the counterpart changes. When a provider assisted Forensic As a Service (Dykstra and Sherman, 2012) is not available, a third way may be considered that is secure, officially supported and reduces the point of contacts with the cloud provider so to possibly shorten times and lower costs. Given the self service nature of cloud platform, object storing is also exposed via entry points that usually reproduce all the features available from a web console. A low level interface based on SOAP or REST web services enables user created applications to remotely execute operations on folders and files such as download and list. Higher level Software Development Kits SDKs are often available that wrap HTTP calls and allow a programmer to rely on languages like Java or PHP. Reasonable scopes of application include, but are not limited to, technical activities performed during pre-trial hearing with or without the consent of the defendant. In the first case the defendant willingly gives his credentials as he may have interest in taking a trusted snapshot of his cloud stored files without any modifications. In the latter scenario, by performing a forensic analysis of a seized computer law enforcement could have recovered username and passwords of a storage account (Quick and Choo, 2013)a or directly an access token string (AT) so to bypass user authentication, as it might be possible for Dropbox (see section 4.6.1.2). Here some issues concerning the applicable jurisdiction may apply, if the cloud platform is located abroad, but the point is disputed. From one side it can be argued that an official legal assistance is due not to acquire data unbeknownst to judicial authority hosting the cloud infrastructure, whereas, from another side, remotely accessing a cloud account by means of client applications which safeguards content integrity and ensures write protection may be considered admissible (Aterno and Mattiucci, 2013) within the umbrella of a local court order only. This uncertainty is most likely to stand until a consolidated case law is established.

While the approach of a remote acquisition seems promising, there are some aspects that need to be deepened before blueprinting strategies and tools able to image remote data in a forensically sound way. First and foremost, forensic best practices, where possible, suggest avoiding alteration of digital evidences (DE) during acquisition. Therefore a read only access to cloud storage areas which mimics the write blocking mechanism applied in traditional bit stream copy of physical mass memories would be beneficial. Indeed, Application Program Interfaces (API) do allow write access: upload, deletion and copy of objects are possible by design. Furthermore, while REST web services seems somehow the "lingua franca" for interacting programmatically with remote storage, the parameters that need to be specified in the calls may vary greatly from one platform to another and so do the format of returned data. An extra layer which harmonizes the syntactic differences is therefore needed. Not less important is the requirement of protecting the integrity of all the retrieved data and reporting all operations in a detailed log. With this foreword, this chapter describes the concepts and internals of the *Cloud Data Imager Library* (CDI Lib), a mediation layer we developed to enforce read only access to files and metadata of selected remote folders, while presenting a unified front end which masks out the syntactic and functional differences of cloud technologies. We built a desktop application on top of the library which, once instrumented with the necessary credentials, provides functionalities like folder listing with view of present, deleted and shared content, browsing of file revisions, extensive logging and imaging of folder trees with export to widespread forensic formats. CDI Lib currently supports access to three popular storage facilities: Dropbox, Google Drive and Microsoft OneDrive.

4.2 Previous and related work

Plenty of work has been developed about discovering traces left on client devices by the interaction with cloud storage platforms. For instance, Chung et al. (Chung, Park, Lee and Kang, 2012) have devised a procedure to collect remnants from computer and smartphones accessing, among others, Amazon S3 and Google Docs and found that many artifacts can be recovered by digging into logs, cache files and databases present in a user profile. In two consecutive papers Quick and Choo (Quick and Choo, 2013)a and (Quick and Choo, 2013)c accomplished a comprehensive analysis concerning traces recoverable in memory and persistent storage of a Windows PCs and Apple iPhone after Dropbox and Microsoft OneDrive services were accessed via browser or client applications. A similar research was accomplished for Amazon Cloud Drive (Hale, 2013). Conversely, procedures and tools for server side acquisition of file content and metadata from a cloud object store appear to deserve a far larger degree of deepening. Quick and Choo again (Quick and Choo, 2013)b have explored the possibility of collecting files from an user account of Dropbox, Google Drive and Microsoft OneDrive. As a preliminary consideration, the authors observe that their investigation lacked a suitable forensic software for the collection of data. As a consequence, their findings are somehow limited by the need of using an internet browser or the official client application. Indeed these are general purpose tools which were not designed with forensic principles in mind and, as the authors themselves observe, may not leave traces in client devices of precious information such a historical versions of files. Considering then the circumstance that one of the ends of the communication is under the control of the researchers, much more could be put in place than capturing SSL encrypted network traffic. However, one of the outcomes of their research is an important starting point of the present work as they determined that there were no changes in contents after having downloaded a file, while only some of the timestamps were preserved. The Cloud Data Imager project just aims at filling the gap outlined by the work of Quick and Choo. A dedicated forensic software could log the full conversation with the cloud platform at application level and in clear text, having if anything the issue to protect user credentials, access and refresh tokens (RT). Furthermore, cloud APIs have provisions for retrieving the metadata of all items in a folder and this is crucial to set creation and modification times of downloaded files equal to the one hosted in the cloud storage area. Concerning the literature relevant to section 4.7, in the case of the Google FS, gaps which need to be filled in the road to a forensically ready cloud storage have been presented in (Spyridopoulos and Katos, 2011). The authors in particular discuss the need for the file system to permanently store the location of servers that host the data fragments composing each file, information which is instead kept in the volatile memory of the master node.

4.3 Personal object stores

Object stores are very popular facilities these days. They allow reliable persistence of user's content like documents or pictures thanks to a sparse architecture able to massively scale and tolerate component failures (Openstack, 2013). Their native interface is based on web services that allow interaction with objects in their entirety: for instance, it is not possible to modify an object by writing a defined amount of bytes at a certain position as allowed by traditional Portable Operating System Interface (POSIX) semantics. As a result, a modification of a document requires its previous deletion followed by an uploading of a new version of it and hence derives the property of immutability (Google, 2014)c. This is an aspect of major interest from a forensics point of view: differently from traditional file systems where an alteration of a file usually leaves no clue concerning its original content, object stores may keep a list of versions of those deleted objects, thus giving the chance to rebuild the history of modifications made to them and possibly disclosing precious information for the digital evidence analyst.

For ease of use, using a provider distributed application, storage areas can also be replicated with a two way synchronization on the file system of user's device and mounted like a regular local folder. An important aspect of a forensic investigation is the possibility for an examiner to find remnants of past activity which are not immediately manifest, not only in the final user's computer or tablet, but also in the cloud infrastructure itself. Data can be consumed from a variety of client devices which might be unavailable for an inspection and may change, be updated or erased: with its outstanding capacity of durability the cloud might be the only anchor of a case. In this respect, object stores usually feature trash containers in which items are put after deletion (Google, 2014)a. Depending on the quality of service subscribed, these can enforce temporary or long term persistence, until users decide for a permanent erasure. TRASH is a system folder to which items are transparently moved awaiting their fate. Equally remarkable for the forensic examiner is the possibility of the cloud platform to keep track of past versions of objects after their are updated by the user. The programming interface which exposes a storage area may have functionalities that go far beyond the retrieval of manifest content and may prove very relevant for computer forensics. For instance, trash and past revisions of a file may be available via remote query, so their accounting may be crucial to an investigation. Finally, not less important, is the possibility to retrieve objects metadata: some "data about data" may be extremely valuable because they are less under user's control. Consider the case in which an examiner could be led astray by a clever suspect that conveniently tweaks file timestamps on his laptop to support his claims. Conversely, uploading a new version of a document to a cloud store, for example as a consequence of a folder automatic backup, may retain its modification time, but updates the creation time to the current remote platform system time so to possibly generate contradictions between the two timestamps. This is likely synchronized with a time server and is thus expected to be a much more reliable landmark than a notebook clock. For completeness sake, it has to be added though that policies on object timestamping may vary from a cloud provider to another and therefore they need to be evaluated case by case.

4.4 Requirements for a novel application

A forensic software, no matter how innovative, needs to comply with concepts and procedures set forth by relevant regulations and best practices. In this respect, this work relies on the guidance offered by ISO/IEC 27037 standard (International Organization for Standardization, 2012). This international standard contains guidelines directed to all the professionals who have to confront with potential DE, from identification to evaluation in a tribunal, needing to grant that this material comply with the generally accepted principles of relevance, reliability and sufficiency. The standard deals with four phases: identification (DE search and recognition), collection (removal of DE from its location), acquisition (creation of a copy) and preservation (safeguards to avoid that DE is tampered with, damaged or dispersed). Even if these guidelines does not explicitly deal with cloud storage services yet, we can consider them as delivered by non interruptible mission critical systems which can be reached only remotely and cannot be acquired in their entirety because of their size. Under these conditions, clauses 5.4.4, 7.1.3.3 and 7.1.3.4 of ISO/IEC 27037 states that a logical partial acquisition which targets specific file and directories is admissible. Focus will be on the two last phases listed by the standard: digital evidence copy and preservation. Identification is considered accomplished a priori as an outcome of an investigative activity leading to pinpoint the relevant cloud accounts.

Collection is not applicable: there is no digital evidence to remove from its location as everything is accomplished via a network. The following paragraph lists the requirements that should be fulfilled by a cloud storage forensic application (FA).

4.4.1 Logical acquisition

Forensic software should leverage provider delivered programming interface that allows unabridged retrieval of file content and metadata irrespective of cloud platform file system technology. An extra effort may be necessary to request by other means data which were not available remotely as in the case of access logs, deleted items or historical versions of documents. Should the provider expose both clear text and encrypted endpoints, the FA should rely on the latter to ensure confidentiality of communications. In this respect, Secure Socket Layer (SSL) is a ubiquitous protocol.

4.4.2 Performed functions

The module responsible for the communication with the cloud platform should implement a minimum set of functions which allow the following operations:

- user authentication and authorization;
- retrieval of user information like name and ID;
- retrieval of folder metadata with its existing subfolders and files. Deleted items should be obtained as well, if possible. Metadata should include at minimum: name, size, creation and modification date.
- listing of all available revisions of file, if available;
- file content download;
- retrieval of directories and files that someone else shared with the user, if this information is provided.

As method invocation syntax varies from a provider to another, this module should provide an harmonization layer which exposes a unified set of calls so to mask out possible differences like Uniform Resource Locator (URL) composition, input parameter list or formatting of returned data.

4.4.3 Low level interface

For completeness sake, developers should exercise care so to select the API that allows to retrieve the maximum amount possible of information. This may mean accessing the platform at the lowest possible level and may require a larger degree of development effort, but at same time it involves remarkable paybacks:

- augmented control, which translates into the possibility of retrieving more potentially interesting information from the cloud platform;
- the possibility to develop an application using languages for whom no SDK exist.

For example, most providers publish SDKs with high level classes that wrap REST web services calls and greatly ease the life of programmers by reducing the amount of code necessary to perform operations on the data store. Consider the case of Dropbox: the latest core Java API to date is version 1.7.3. Invoking a method called *getMetadataWithChildren* from class *DbxClient*, which accepts the path of a folder as input, a list of entries of type *DbxEntry.File* or *DbxEntry.Folder* is returned, that represents all the files and subfolders contained within. Each entry is a data structure which does not contain a flag to inform if the folder or file has been deleted. The same happens with the latest Java API for Android version 1.5.4, where folder listing can be obtained by calling a method named metadata belonging to class *DropboxAPI*. This time the returned list of structures, namely *DropboxAPI.Entry*, would include a field named *is_deleted* which however is not assigned because metadata has no input parameter to request the inclusion of deleted items in the returned list. Getting those items from Dropbox could be possible as they live for 30 days for unpaid accounts and forever in case of paid subscriptions. It is therefore necessary to leverage the REST web services interface, which is the foundation of every higher level SDK: invoking a GET method with an *include_deleted* parameter equal to true returns a JavaScript Object Notation (JSON) formatted list which includes deleted folder and files. It was the only way we were able to achieve this result.

4.4.4 Read only access

Conforming to the principle of reliability of digital evidence, cloud stored content and metadata should be secured against any accidental change. This translates into the requirement for the FA to access remote content in a read only manner. Indeed, similar to traditional bit stream copies of digital evidences, best practices advise, if possible, to implement a write blocking mechanism to avert the risk of invalidating an acquisition. This cannot always be guaranteed in case of usage of internet browsers and provider delivered client applications because they could possibly cause accidental modifications to storage areas. Other avenues for alteration of the remote content which could be performed by third parties must be discussed with the provider and eliminated, if possible. Example solutions could include:

- a new account released to LE with exclusive access to suspect's storage area;
- if suspect's recovered own credentials are used, exclusive login could be granted only to LE's forensic workstation or write permissions could be removed by the provider from the account.

4.4.5 Officially supported interface

The forensic application will be mandatorily based on stable and officially supported API. Under no circumstance programming interfaces offered by third parties can be leveraged, if they did not received a prior endorsement by the cloud provider. This strongly excludes for example function calls which were obtained by reverse engineering of code or via protocol inspectors.

4.4.6 On demand folder browsing

Conforming to the principle of sufficiency of digital evidence, the FA should offer the possibility to browse an account in order to possibly exclude from imaging those folders that appear clearly irrelevant. To avoid unnecessary network traffic, instead of walking the whole directory tree beforehand, metadata can be retrieved and cached only when examiner's navigations requests it. The chance of performing a prior selection is also important for triaging data in case of very large stores that cannot be wholly acquired in the allowed timeframe. File content preview should be possible also for deleted and previous versions, if available. The on demand nature of folder browsing excludes a blind synchronization of the data store with a local folder when the FA starts.

4.4.7 Native logging

Clause 5.3.2 of ISO/IEC 27037 states the importance of documentation to allow an independent assessor to evaluate all actions performed. To meet this requirement, the forensic software should therefore support a logging facility of configurable verbosity to create an audit trail for all actions. All relevant events stemming from the interaction with the cloud platform, user actions or error conditions should be accounted for. The times should indicate the shift from Coordinated Universal Time (UTC), if any, for disambiguation purposes. If possible, at the end of operations, the log file should be hashed and timestamped via certification services delivered by a legal authority in order to locate it in a defined point in time. Due care should be observed in protecting sensitive information like user credentials or restricted configuration parameters. For instance, when a request of listing the content of a folder is issued, access tokens used for authorization get recorded. Protection could be enforced by creating an unabridged master copy of the log file which is stored securely and a working copy to be delivered to trial parties where sensitive data are masked off. This way original information can be accessed in a controlled manner should the Court deem this necessary. An efficient logging methodology could avoid the usage of extra recording facilities like screenshots, video footages or secure HTTP protocol decoders as Telerik's Fiddler web proxy[xli], which is able to decrypt protected traffic with a man-in-the-middle approach.

4.4.8 Folder imaging

Once the examiner has selected the relevant folder, the FA must faithfully traverse the complete tree so to copy remote data into logical evidence files and should compute integrity protection codes to avert the possibility that the evidence is tampered with after it has been acquired. For instance, a cumulative cryptographic hash of all data retrieved by the server could be calculated and an accompanying file hash list could be a valuable addition. The output format of the image may vary: it could be for instance a database or a local folder that reproduces the structure of the cloud area. In the latter case, metadata information can be put in a text file inside its corresponding folder: these are the trusted origin of information about files and folders, in particular when copied items do not preserve some fields such as creation time. The crucial aspect of imaging operations is that file contents and metadata of the target folder and subfolders be copied in their entirety as received by the server, including trashed files and revisions. The latter may constitute an added value over analyzing physical copies of disks: for instance, in a New Technology File System (NTFS) formatted volume of a Windows 7 box, previous versions of a file are recoverable only if a user made a backup or if System Protection is enabled to allow shadow copies be created. Even in this case, only the copy which was present just before the restore point creation will be available.

4.5 Evaluation of current tools

After a literature perusal and due technology scouting, it appears that the arsenal for remote data retrieval of cloud storage areas is not very populated. General purpose tools such as internet browsers or provider delivered client applications can be useful allies in an investigation and can be certainly used if they produce the expected results, but we need to stress an incomplete compliance to some of the above requirements, for instance read only access or native logging. They are not forensics applications indeed, but were designed to allow a convenient read-write access to users, so missing functions need to be provided externally by other software. Provided that he is able to justify the reason for his actions, the investigator is not bound to specific tools, but it could be beneficial to the quality of technical assessments relying on instruments able to increase the overall level of auditability and justifiability. The following table reports a compliance test to the aforementioned requirements in case of access by means of browsers (Microsoft IE10 and Mozilla Firefox 25.0.1) and desktop applications for Dropbox (version 2.0.22), Google Drive (release 1.12.5329.1887) and Microsoft OneDrive (build 17.0.2015.0811). Although coarse grained, this test brings some food for thought.

Requirement	Client	Browser	Notes
Logical acquisition	Pass	Pass	Both tools allow to create local copies of remote files and folders
Performed functions	Fail	Pass	Browser access is compliant with all requested functions, conversely no deleted items and previous versions of files available with all desktop clients. No shared contents available for Google Drive and Microsoft OneDrive clients
Low level access	Fail	Pass	Considering the rationale of retrieving the maximum amount possible of information, desktop clients fail as they do not show versioned or deleted files, which is instead possible with a browser access
Read only access	Fail	Fail	Write access is granted for both tools
Official interface	Pass	Pass	Desktop clients are delivered by cloud providers themselves and browsers access provider web sites
Folder browsing	Fail	Pass	Browser allow on demand folder navigation. Client applications imply prior blind synchronization of the remote data store, which may include irrelevant data
Native logging	Fail	Fail	Neither tool logs communications with remote servers with the needed detail. Extra recording tools can be put in place, such as screen captures and video recording of operations. For browsers, tools like Telerik's Fiddler web debugging proxy can be leveraged which decodes secure HTTP traffic
Folder imaging	Fail	Fail	Both can acquire a whole directory tree, but some folder metadata is not preserved. No hash functions are used to protect integrity for both tools

Table 4-1 Fail/pass test for browsers and client applications

Table 4.1 shows that desktop clients, despite allowing a convenient navigation on a local copy of data, do not show objects that were revised or deleted, do not preserve item creation time and by default perform a blind synchronization, possibly including irrelevant material. An internet browser has more to offer from a forensic point of view as it can download selected folders as compressed files or show deleted items and revisions (in Microsoft OneDrive the latter are available only for Office documents). However, both tools allow modification of target items and do not provide dedicated logging or integrity protection via hash codes. As a part of its forensic products offer, F-Response[xlii] delivers a Cloud Connector which enables one to mount a cloud storage platform as a local logical volume or network share. Unfortunately, a copy of the software is not available for tests, but it is anyhow worth noting that, according to the public available manual rel. 5.0.1, it is guaranteed a write protection mechanism. This is compliant with clause 4.4.4 and is very important from a forensic standpoint. However, performed functions seem not to include the possibility to retrieve deleted items, past versions of files or shared folders. These features, according to contacts with the company, will be provided in future releases. Differently from the provisions of clause 4.4.8, Cloud Connector does not directly perform folder imaging, but rather prepares the ground for a third party product.

4.6 Architecture and functions

Cloud Data Imager is a novel forensic tool for the remote collection of data from cloud storage accounts which fulfills all the requirements listed in the previous sections. The two main features are directory browsing with visualization of file content and logical copy of a selected folder tree, not necessarily the root, to a local repository. Access to Dropbox, Google Drive and Microsoft OneDrive platforms is currently implemented, but development plans include support for other "Storage as a Service" facilities either public such as Amazon S3[xliii] or private like Openstack's Swift[xliv]. The tool features a library which mediates between an overlying application and the provider exposed programming interface. Figure 4.1 shows global architecture and functions:

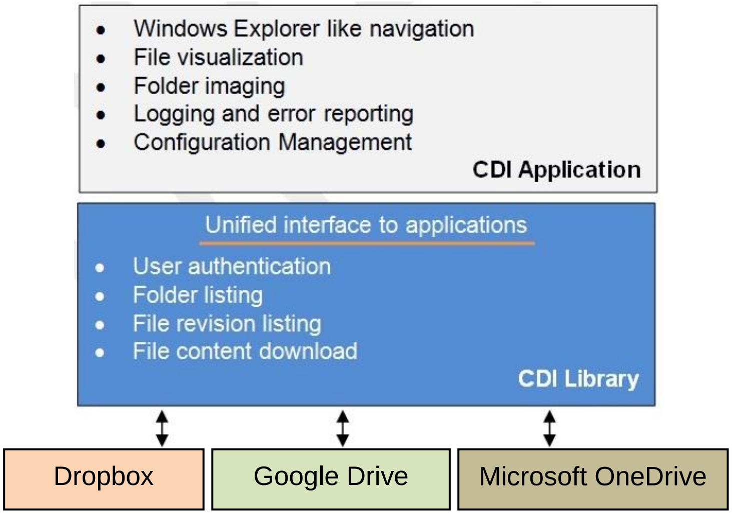


Figure 4-1 CDI architecture and functions

4.6.1 CDI library

The APIs exposed by the mentioned providers have some important commonalities such as an interface based on HTTPS requests and the usage of OAUTH 2.0 as a protocol for authentication and authorization (Hardt, 2012). However, there are important differences that must be accounted for: URLs to which direct requests, methods and syntax for parameter passing and data structures returned in server answers. After being initialized with a provider ID, the library hides this lack of homogeneity and publish an unified set of calls irrespective of the underlying cloud technology. This enforces interoperability among cloud platforms and greatly simplifies the development of an application built on top of the library as a distinction among providers is made only once in the part of code that handles initialization. The available functions are listed in table 4.2.

Name	Category	Function
getAuthorizeUrlV2	Authentication and Authorization	OAUTH 2.0. Retrieves the URL to be addressed by a browser to let user authorize access to cloud account. If user authorizes, page returns an authentication code
getAccessTokenV2	Authentication and Authorization	OAUTH 2.0. Exchanges the authentication code in the authorization web page to an access token string to be used in subsequent service request
getAccountInfo	Information	Retrieves user name and ID of the account holder
listFolder	Browsing and Imaging	Retrieves metadata of a folder including its files and subfolders
listFileRevisions	Browsing and Imaging	Retrieves metadata of all previous revision of a file
getFileContent	Browsing and Imaging	Gets the raw content of a file

Table 4-2 List of calls exported by CDI library

It can be seen that requirement 4.4.2 is satisfied along with 4.4.3, because CDI leverages the REST web services interface which is at the lowest possible level. Read only access (see requirement 4.4.4) is then guaranteed as all methods which handle users data are based on HTTP GETs. The library was developed after browsing the official literature (requirement 4.4.5) published by Microsoft (REST reference Live Connect, 2014), Google (Google, 2014)b and Dropbox (Dropbox, 2014). Finally, operations get logged in a text file with a configurable verbosity level which defaults to DEBUG, the most complete (requirement 4.4.7). This means that all the requests and responses generated by functions listed in table 4.2, saved *getFileContent* to avoid excessive space consumption, are recorded verbatim as issued to or received from the cloud platform. The CDI project is written in C# for .NET framework 4 or higher and requires Windows 7 or later. It leverages Json.NET package (Newton-King, 2013) to parse JSON formatted server answers along with Log4net (The Apache Software Foundation, 2013)b as a logging facility. All the providers listed in the following require that a developer register to get an app key and secret. These credentials are embedded in CDI library authentication and authorization functions.

4.6.1.1 OAUTH

The OAUTH authorization framework version 2 is the de facto standard for authorizing access via web services (Hardt, 2012) to a restricted resource on behalf of a third party. Differently from traditional client-server scenarios, a client application is not aware of the credentials of the owner: it is issued a temporary token to access the resource after the owner has logged on an authorizing server and explicitly accepted the access scope requested by the application. Scopes may include for instance the permission to modify an entire directory tree or just a single folder and the ability to operate when the owner is not logged on the storage platform. Access tokens are character strings of variable length issued by authorization servers which need to be attached to authenticate every request to resource servers. They are usually short lived: one hour is a typical value. To allow offline operations when the user is not logged in, a refresh token may be released to applications along with the access token after a user has given his consent. RT's goal is to be presented to authorization servers only to acquire a new AT so to extend admittance without requiring user intervention. In this respect, CDI library keeps track of AT lifetime and, prior to expiration, leverages RTs to transparently renegotiate the issue of a new one. This offloads the application from handling the renewal of credentials from time to time. Figure 4.2 shows OAUTH 2 flow triggered by CDI over an HTTPS protected channel:

1. a browser session is started where the Authentication & Authorization server is contacted and a login form is presented to the user;
2. user authenticates and approves the list of scopes requested by the application;
3. Authentication & Authorization server releases an AT and possibly a RT to allow the application to operate on behalf of the user when he is not logged in;
4. The AT string is attached to any following resource request.

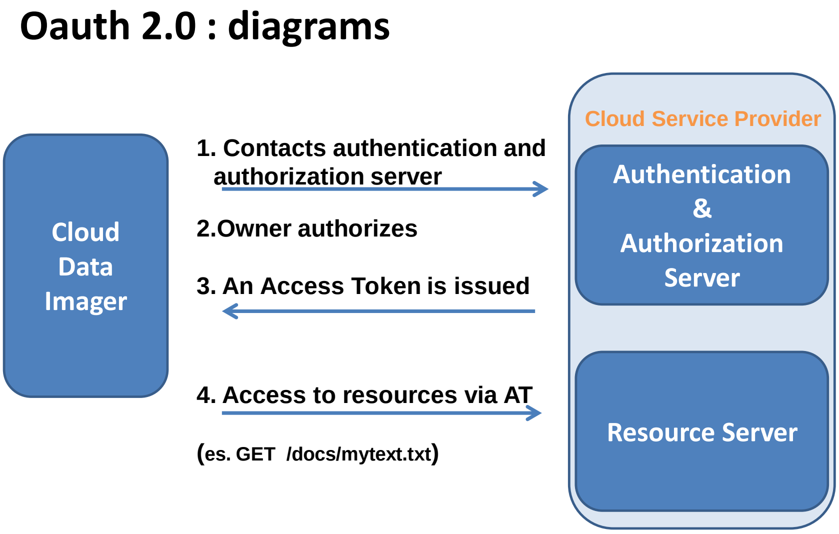

Oauth 2.0 : diagrams
Cloud Data Imager
1. Contacts authentication and authorization server
2.Owner authorizes
3. An Access Token is issued
4. Access to resources via AT
(es. GET /docs/mytext.txt)
Cloud Service Provider
Authentication & Authorization Server
Resource Server

Figure 4-2 CDI Oauth 2 flow

4.6.1.2 Dropbox

Dropbox's authorization scopes include read/write permissions on either a dedicated folder or the full user's dropbox. It's a coarse grained permission scheme as, for instance, it is not possible to get read only access to all files and folders, even if content integrity is still preserved thanks to http GETs usage made by CDI library. Differently from other providers, access tokens have not an expiration time: they can be used in the long term until user repeats authorization process or explicitly revokes them. This means that once stored on a durable medium, a poor protection policy of the AT could lead third parties to bypass user authorization and access his data without further ado. For this reason CDI library keeps AT only in volatile memory and, as stated in paragraph 4.4.7, in the log file asterisks are put in place of AT characters to avoid sensitive information leaks. Listing of a folder items and file revisions has an upper bound. The former defaults to 10,000 and the latter to 10. An error code will be returned for listings containing a number of files exceeding the limit. Accordingly, *listFolder* and *listFileRevisions* will return no more than 25000 and 1000 files which are the topmost listing limit. Dropbox's allows retrieval of deleted and revised items: unlimited deletion recovery and version history is granted to paid accounts whereas this ability is limited to 30 days for the free ones. Concerning the metadata returned by the API for deleted files, it has been verified that their size is zero bytes and *client_mtime*, the original file modification time which is retained if the file is uploaded with Dropbox's desktop application, is invariably set to Dec 31st 1969, 23:59:59 +0000.

4.6.1.3 Google Drive

Google Drive has a more flexible authorization scheme with a granularity ranging from full read/write permissions on all user files to single per file access. CDI library leverages "*https://www.googleapis.com/auth/drive.readonly*" parameter, which grants read only access to all files and metadata thus giving further assurance that user data are not modified in any way. Access tokens have a typical lifetime of one hour and are issued along with refresh tokens because CDI library requires an offline access type in order to carry possibly lengthy calculations such as folder imaging. Google Drive keeps track of file versions, but this requires space. So they are deleted after 30 days or if there are more than 100 revisions of a file. However, the user can decide to avoid this auto deletion policy on a per file basis. Listing of a folder defaults to 100 items and *listFolder* will stretch to the upper bound of 1000 items. Conversely, there are not input parameters which limit the number of returned revisions of a file, even if the default number will be 100 as per deletion policy.

4.6.1.4 Microsoft OneDrive

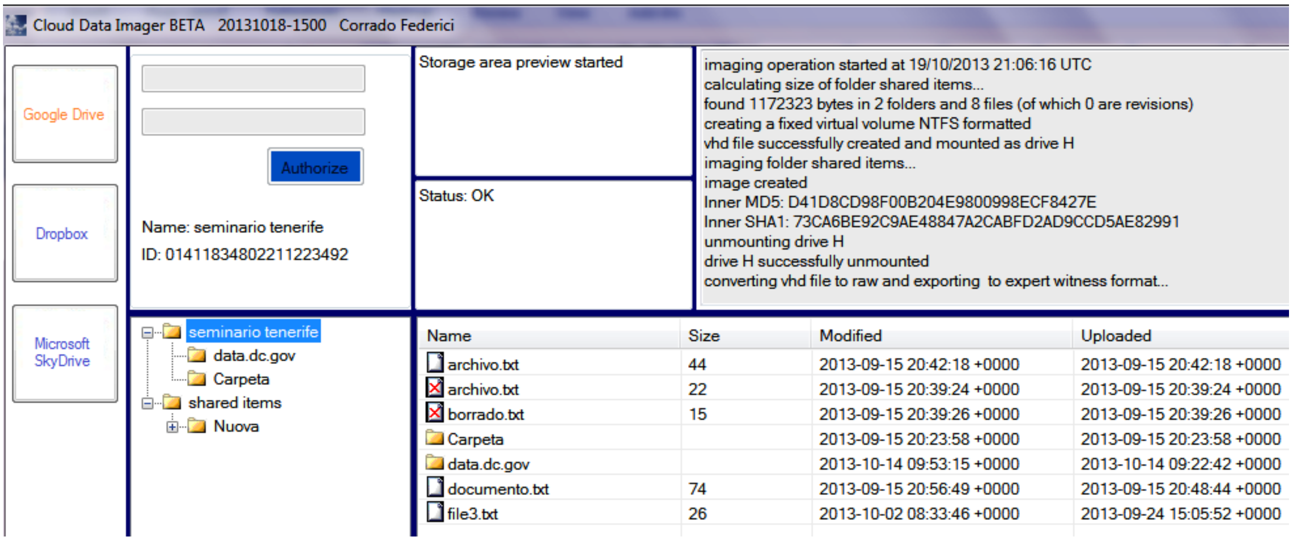
Microsoft OneDrive features an even more comprehensive authorization scheme which is able to give separate permissions to user's profile, contacts, calendar, multimedia content or more generally to files. CDI library uses the following scopes:

- *wl_basic*: to get user's name ad ID;
- *wl_contacts_skydrive*: to obtain read only access and retrieve metadata and content of all folder and files belonging to the users or shared by others;
- *wl.offline_access*: to operate also when the user is not signed in via the refresh token mechanism.

Once the user has authorized, scopes are cached in the "App and services" tab of his account and need to be explicitly revoked in case of need. Deleted files are sent to the recycle bin and kept for at most 30 days in case of free accounts. Permanence in the bin depends on its size: if it reaches 10% of the storage capacity files are removed earlier, but not before three days after deletion (Shahine, 2012). Version history exists, but they are available for Microsoft Office documents only. The most severe limitation from a forensic standpoint is the lack of API functions that expose the recycle bin and previous releases of a file. This somehow weakens the power of remote collection tools, even if the benefits of ensuring a write protection, producing an audit trail and safeguarding integrity via hashes is still rather valuable. The API call for listing folder contents has no input parameter limiting the number of returned children.

4.6.2 CDI Application

The dashboard is divided into three functional areas as depicted in figure 4.3: a central area with a tree and list view for navigation purposes, a left panel for provider selection and an upper zone for information. The tree shows the selected folder only, whereas the columns in the list show items Name, Size, Modification and Upload date. The upper right zone details diagnostic information as logged by the application, starting from the selection of a provider: every action such as folder listing or visualization of a file is recorded in a session log whether successful or not. In the latter case the error code returned by the cloud platform is written as well.



Cloud Data Imager BETA 20131018-1500 Corrado Federici

Name: seminario tenerife
ID: 01411834802211223492

Storage area preview started

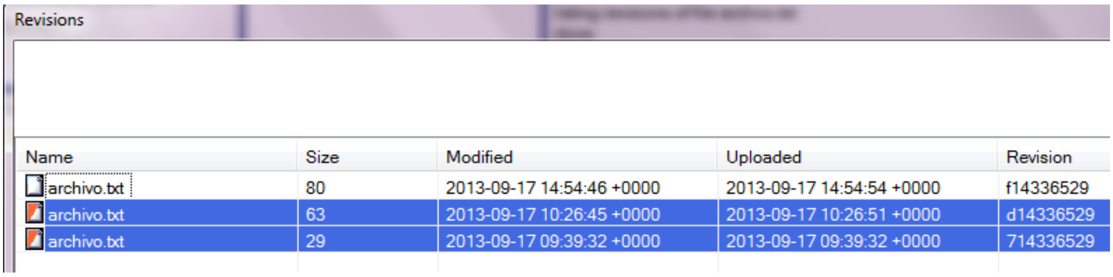
Status: OK

```
imaging operation started at 19/10/2013 21:06:16 UTC
calculating size of folder shared items...
found 1172323 bytes in 2 folders and 8 files (of which 0 are revisions)
creating a fixed virtual volume NTFS formatted
vhd file successfully created and mounted as drive H
imaging folder shared items...
image created
Inner MD5: D41D8CD98F00B204E9800998ECF8427E
Inner SHA1: 73CA6BE92C9AE48847A2CABFD2AD9CCD5AE82991
unmounting drive H
drive H successfully unmounted
converting vhd file to raw and exporting  to expert witness format...
```

Name	Size	Modified	Uploaded
archivo.txt	44	2013-09-15 20:42:18 +0000	2013-09-15 20:42:18 +0000
☒ archivo.txt	22	2013-09-15 20:39:24 +0000	2013-09-15 20:39:24 +0000
☒ borrado.txt	15	2013-09-15 20:39:26 +0000	2013-09-15 20:39:26 +0000
Carpeta		2013-09-15 20:23:58 +0000	2013-09-15 20:23:58 +0000
data.dc.gov		2013-10-14 09:53:15 +0000	2013-10-14 09:22:42 +0000
documento.txt	74	2013-09-15 20:56:49 +0000	2013-09-15 20:48:44 +0000
file3.txt	26	2013-10-02 08:33:46 +0000	2013-09-24 15:05:52 +0000

Figure 4-3 CDI dashboard

There are a few configuration parameters, some belonging to a common part and some differentiated, included in a XML file. In particular, there are two lines in which application key and secret are recorded. This is because Cloud Data Imager is in beta version to date and has not been yet endorsed by cloud providers which may enforce limits on access for applications not yet ready for production. For instance, Dropbox allows at most one hundred concurrent accesses for applications still under development, so the author cannot embed his credentials for widespread diffusion yet. Therefore, CDI users will have to register a fictitious application for every provider for which no official approval exist and get application credentials. These need to be input only once for provider in the edit boxes just above the blue "Authorize" button (see fig. 4.3). Once written in the configuration file their confidentiality is protected leveraging Effortless.NET encryption library (Effortless .Net Encryption, 2012) with a 256 bit key generated from a user chosen passphrase. After a cloud provider has been selected, a new work session can begin and a check is performed against the presence of a valid AT in the configuration file. Recalling the introduction, this could be when a technical activity is carried out bypassing user consent when investigators have to enforce a court order. In this case, an AT could therefore have been directly obtained by the provider or retrieved as a result of the inspection of a suspect's equipment, but must not have a limited lifetime because no out of band refresh token is expected in the configuration file. Otherwise, if no AT is available, the whole OAUTH 2.0 process is started: a browser session is initiated which requests authentication to the selected cloud platform. Once user has successfully logged in, an authorization page is presented that states the access scopes requested by the application. Once the user has accepted, his ID is retrieved along with the content of the root folder, which is named after the user ("*seminario tenerife*" in Fig.4.3) and the content of the items which have been shared with the user ("*shared items*" in Fig.4.3). This latter is not shown for Dropbox because shared folders appear as root children. From now on the usual explorerlike navigation can begin. As stated in paragraph 4.4.6, the metadata of files and subfolders are retrieved only on demand and then cached. A file can be visualized by double clicking on it. This action has the file downloaded, saved in a configurable working directory and opened with the associated viewer. Deleted files, if supported, are marked with a red X and are viewable as well. Also file versions, if available, are viewable with a right click on a selected file and their icon has the left side filled with red. The list is displayed in a separated window, with the newest release on top, which shows a revision ID (Fig.4.4).



Name	Size	Modified	Uploaded	Revision
archivo.txt	80	2013-09-17 14:54:46 +0000	2013-09-17 14:54:54 +0000	f14336529
archivo.txt	63	2013-09-17 10:26:45 +0000	2013-09-17 10:26:51 +0000	d14336529
archivo.txt	29	2013-09-17 09:39:32 +0000	2013-09-17 09:39:32 +0000	714336529

Figure 4-4 An example of Dropbox file revisions list

4.6.2.1 Imaging a directory tree

With a "right click and confirm" on a folder in the tree navigation pane the user can unleash the imaging process which, in accordance with paragraph 4.4.8, entails a logical copy of metadata and content of every subfolder and file. This is a three stages run:

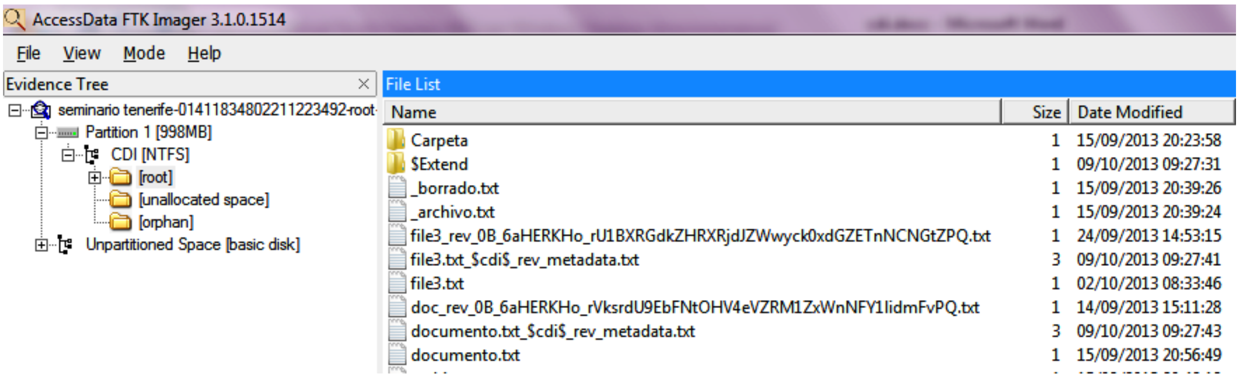
1. creation, format and mount of a virtual hard disk (VHD) which will host the logical image. VHD is a specification made public by Microsoft under its Open Specification Promise (Microsoft, 2013) for encapsulating a volume in a file. By leveraging a hidden instance of the Diskpart Windows utility, CDI creates a fixed virtual hard disk, whose room requirements are calculated in a preliminary phase according to cloud storage size plus a 20% margin. In any case, the minimum volume size is 512 MB to keep low the percentage of sectors requested by file system service structures compared to space

available for data. Alternatively, if the user knows the upper bounds of cloud storage size, he can choose a predefined or custom virtual hard disk size so to save the time needed to calculate remote storage dimensions. In all cases the virtual volume will be NTFS formatted to overcome FAT32 4 GB file size limitations and it is possible to decide whether to perform a quick or a full format. At last the volume will be mounted and assigned the first available drive letter from H to Z;

2. At this point the whole remote directory structure is recreated on the mounted drive. For each folder, a text file named "cdi_metadata.txt" is created which contains the unabridged server response to the *listFolder* call. For each file owning at least one historical version, a text file named after it is created to whom the suffix "_cdi_rev_metadata.txt" is appended. It again contains the complete answer to the invocation of the *listFileRevisions* function. These additional files are therefore created by the imaging process and their goal is clear: as they contain the metadata of every folder and file present in the cloud storage, their presence mimic the acquisition of the remote file table structure. Creation and modification timestamps of every recreated item are set equal to the original, but this is just for reader's ease. In case of doubt, trust must be put only in the content of xxx_metadata.txt files. For all data received by the server, two cumulative message digests are calculated with MD5 and SHA1 cryptographic functions and recorded in the log. These will be known as **inner hashes** and will be checked against the content of all downloaded files plus xxx_metadata.txt files. An MD5-SHA1 hash list of all files is also created. To verify inner hashes, the imaged folder needs to be traversed in the exact order of this list otherwise there will be not match. When the list is produced, a check is made to compare the calculated

value against the expected hash value included in file's metadata, if available (as in the case of Google Drive which stores an md5 hash of each file). An error is reported in the log if a mismatch is detected;

3. At last, the fixed virtual disk is dismounted and the .VHD file is converted to raw format by removing a 512 bytes footer. It is then exported in the Guidance Software's *Expert Witness Format*, one of the most widespread forensic container to date, silently running the *ewfacquire* tool from the *libewf* (Metz, 2013) project. EWF is a compressed format so the outcomes are .Exx files whose sum can be much less in size that the virtual disk. EWF has provisions for editable additional information such as case number or notes. Values of inner hashes are appended to these notes for examiner's convenience. *Ewfacquire* will also include in the container an MD5 and a SHA1 message digest calculated on the whole virtual volume so to preserve its integrity. This includes NTFS service tables and files and therefore will be definitely different from the inner hashes. They will then be called **outer hashes**. Figure 4.5 displays the content of a sample EWF file, created as the result of the image of root folder in figure 4.3, opened with AccessData FTK Imager utility[xlv] with which outer hashes can be verified.



seminario tenerife-01411834802211223492-root
Partition 1 [998MB]
CDI [NTFS]
[root]
[unallocated space]
[orphan]
Unpartitioned Space [basic disk]

Name	Size	Date Modified
Carpeta	1	15/09/2013 20:23:58
$Extend	1	09/10/2013 09:27:31
_borrado.txt	1	15/09/2013 20:39:26
_archivo.txt	1	15/09/2013 20:39:24
file3_rev_0B_6aHERKHo_rU1BXRGdkZHRXRjdJZWwyck0xdGZETnNCNGtZPQ.txt	1	24/09/2013 14:53:15
file3.txt_cdi_rev_metadata.txt	3	09/10/2013 09:27:41
file3.txt	1	02/10/2013 08:33:46
doc_rev_0B_6aHERKHo_rVksrdU9EbFNtOHV4eVZRM1ZxWnNFY1lidmFvPQ.txt	1	14/09/2013 15:11:28
documento.txt_cdi_rev_metadata.txt	3	09/10/2013 09:27:43
documento.txt	1	15/09/2013 20:56:49

Figure 4-5 FTK imager's view of root folder

Making a comparison to figure 4.3, it can be seen that:

- conforming to the convention adopted by the most famous tools as Encase, deleted items are prepended with an underscore;
- file3.txt has one previous version "file3_rev_0B_6aHERKHo_rU1BXRGdkZHRXRjdJZWwyck0xdGZETnNCNGtZPQ.txt" named after it with its revision id suffix. "file3.txt_cdi_rev_metadata.txt" contains revision information as received by the cloud storage platform;
- the same applies to documento.txt, saved the fact that it was renamed from doc.txt;
- modification dates of all downloaded files and folders are retained. Creation timestamps, that is date and time an item was uploaded to the cloud, are retained as well, but are not showed by FTK Imager. A search for these date and times can be made against the content of file "cdi_metadata.txt" which is not shown for the sake of brevity;
- volume is named CDI.

It should be clear at this point that EWF files produced by CDI are functionally equivalent, for the part of file system content/metadata and neglecting slack space and unallocated sectors, to bit stream imaging a physical hard drive NTFS formatted containing present and deleted files/folders hosted in a cloud personal storage. Outer hashes protect EWF files integrity and chain of custody preservation whereas a widespread format guarantees that every forensic expert worldwide is able to handle images produced by CDI.

4.6.2.2 Test findings

We have devised a field tests plan organized in one hundred trials, half of which were accomplished to put under stress the application and half to try out all functionalities with small collections. In the former scenario, all runs but two were successfully carried out to verify operation continuity beyond the hour of imaging activity when an access token is silently renewed by CDI library (not applicable for Dropbox). We hashed every sample file before uploading it to a folder in the cloud storage and repeated the same operation for its corresponding copy after each session. We detected no differences in content between original files and copies after imaging multi gigabyte folders with thousands of files of many different kinds, like pictures, videos or documents, some weighting several hundred of megabytes. The two failed runs were caused by network issues and just required a fresh restart. Error codes were displayed on the screen and recorded in the session log file. In future releases of the tool we will consider the possibility to resume operations from the point of interruption. Functionality tests were all successful. For instance, in a session a few public documents belonging to the data catalog of the District of Columbia (http://data.dc.gov/) were downloaded and unzipped, notably Crime Incidents from 2011 to 2013 and Purchase Orders from 2008 to 2011. These were selected because they are easily editable to verify how CDI wields historical versions. We calculated an MD5 and a SHA1 hash for every file using *HashCheck* Windows shell extension version 2.1.11[xlvi] and created with this tool two separate hash lists, namely data.dc.gov.md5 and data.dc.gov.sha1, to simulate the presence of small sized text files. The 9 files collection hosted in a folder named *data.dc.gov* is listed in table 4.3. We carried out all operations with a Windows 7 box on October 14th 2013 UTC +2, which coincides with the creation date of all files (not showed), selecting a data connection ranging in the average from 200 to 300 Kbytes/sec to verify CDI response with low speed networks. However, such a connection already suffices for a positive user experience as navigation usually entails acceptable delays in opening folders.

Name	Last mod	Size	MD5	SHA1
crime_incidents_2011_CSV.csv	13/10/2013 8:40	5136 KB	7e1854bcb6ebe6267650b49e88c43860	867fba1fb14864aeb4826e4f20a5c5e4fe91704b
crime_incidents_2012_CSV.csv	13/10/2013 8:35	5323 KB	688047a9186a8254d3e01c374e47b3d8	55efdc1097451fb1ff332ab4fd248639760040f2
crime_incidents_2013_CSV.csv	13/10/2013 8:30	4092 KB	afb6d9a2a2ecbca577586cf41284b775	75716055e36e1ddc5ba7ba92ce47f4c82f2230c9
data.dc.gov.md5	14/10/2013 10:21	1 KB	590791fb8e75c6b213b4be36c0753cd0	25096eb28b031e2a9595d0251d716dbc7162518f
data.dc.gov.sha1	14/10/2013 10:21	1 KB	796b74ebc114b0486550f787d5fe93ed	3c2c784649f21af943230450ed70fea68ec6c421
pass_2008_plain.xml	31/12/2009 3:50	15581 KB	4666f0b18baf059a5f4acdcef0217e0d	8b0c6cd118665c7ed662ef0d0ae7bb37fcc867f3
pass_2009_plain.xml	31/12/2010 3:50	14531 KB	0669a81a8d6b6681dcfb444df3428ab5	aa09ee3ff3708d34559d30db7c65f350e38557dd
pass_2010_plain.xml	31/12/2011 3:50	13659 KB	fe35c7d30ad6f2ebcfae50b3bbcc2cf8	ad5b29495376da30e52ba10b17294c984d97ebed
pass_2011_plain.xml	31/12/2012 3:50	12096 KB	07d3e48f971bd0f4bb40a7e7c32bbfa7	c2778fbaf497938efe730e8efb9a74dfc947a555

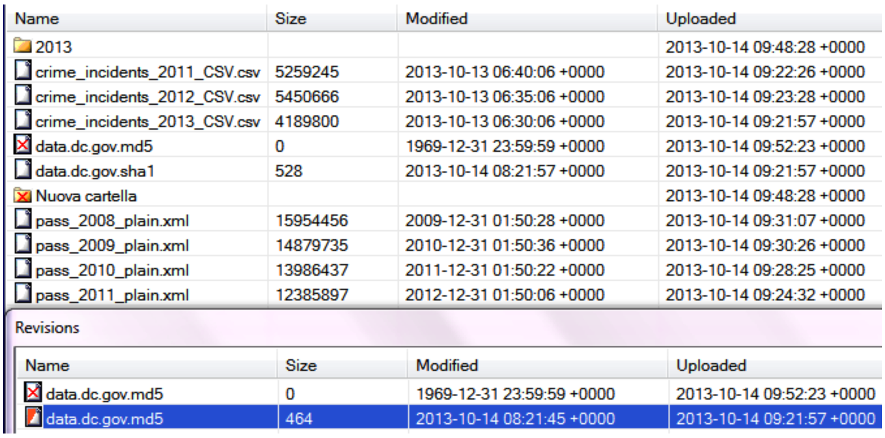
Table 4-3 List of the sample collection used for testing

We uploaded the folder *data.dc.gov* to Dropbox, Google Drive and Microsoft OneDrive leveraging the associate desktop application. At a later time, we moved the file data.dc.gov.md5 to a newly created subdirectory named "*2013*" and then edited it to create a revision by removing all asterisks from the content, obtaining the new hash values of 5c87472d77ff352f20469baf4648918a and 296a78be66c2c727338abdbbd57606ce35f3b280 for MD5 and SHA1 algorithm respectively. We performed these operations also for OneDrive, notwithstanding the mentioned inability of remotely retrieving past versions and trashed items. Findings of imaging activity of root folder *data.dc.gov*, containing 9 files and 1 subfolder were as follows:

- **Dropbox**: The imaging process took 5 minutes and 50 seconds, inclusive of remote folder size calculation, to create a 9587 KB .E01 file, discovering and downloading twelve files and 2 subfolders for a total amount of 72108148 bytes. There are 3 files and 1 folder more than the Windows local folder (see Fig 4.6 and 4.7). This is because data.dc.gov.md5 is still accounted as a zero sized file in the root and its revision stems from the uploading of file in table 3. In subfolder "*2013*" there is the last edited copy plus its revision generated from the moving of file data.dc.gov.md5. So in all there are two revisions plus a zero sized file in excess. Explanation of the sequence may clarify further:

1. At 09:21:57 UTC (server time) data.dc.gov.md with modification time of 08:21:45 UTC is created in the cloud storage root after client synchronization;
2. At 09:52:23 UTC the file is moved to subfolder "2013" and a zero sized file is created.
3. At 09:53:49 UTC (client time) the content is changed as asterisks are removed in local copy. A new synchronization forces the creation of a new remote object at 09:54:12 (server time).

The excess folder is a deleted one named "*Nuova cartella*" (New folder) which is the original name assigned by Windows before renaming to "*2013*". The imaging process created five more files: three to host metadata of all directories ("*root*", "*2013*" *and* "*Nuova cartella*") plus two for revisions metadata of data.dc.gov.md5 in folder "*root*" and "*2013*".

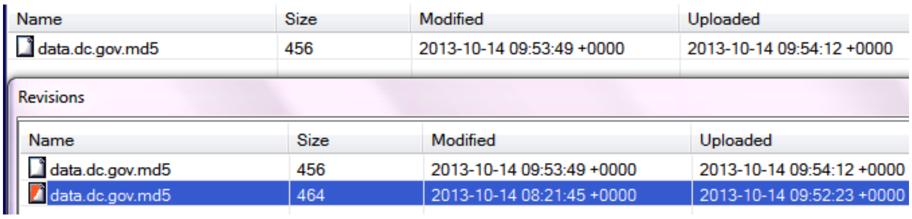


Name	Size	Modified	Uploaded
2013			2013-10-14 09:48:28 +0000
crime_incidents_2011_CSV.csv	5259245	2013-10-13 06:40:06 +0000	2013-10-14 09:22:26 +0000
crime_incidents_2012_CSV.csv	5450666	2013-10-13 06:35:06 +0000	2013-10-14 09:23:28 +0000
crime_incidents_2013_CSV.csv	4189800	2013-10-13 06:30:06 +0000	2013-10-14 09:21:57 +0000
data.dc.gov.md5	0	1969-12-31 23:59:59 +0000	2013-10-14 09:52:23 +0000
data.dc.gov.sha1	528	2013-10-14 08:21:57 +0000	2013-10-14 09:21:57 +0000
Nuova cartella			2013-10-14 09:48:28 +0000
pass_2008_plain.xml	15954456	2009-12-31 01:50:28 +0000	2013-10-14 09:31:07 +0000
pass_2009_plain.xml	14879735	2010-12-31 01:50:36 +0000	2013-10-14 09:30:26 +0000
pass_2010_plain.xml	13986437	2011-12-31 01:50:22 +0000	2013-10-14 09:28:25 +0000
pass_2011_plain.xml	12385897	2012-12-31 01:50:06 +0000	2013-10-14 09:24:32 +0000

Revisions

Name	Size	Modified	Uploaded
data.dc.gov.md5	0	1969-12-31 23:59:59 +0000	2013-10-14 09:52:23 +0000
data.dc.gov.md5	464	2013-10-14 08:21:45 +0000	2013-10-14 09:21:57 +0000

Figure 4-6 Dropbox's content of data.dc.gov and revisions of file data.dc.gov.md5



Name	Size	Modified	Uploaded
data.dc.gov.md5	456	2013-10-14 09:53:49 +0000	2013-10-14 09:54:12 +0000

Revisions

Name	Size	Modified	Uploaded
data.dc.gov.md5	456	2013-10-14 09:53:49 +0000	2013-10-14 09:54:12 +0000
data.dc.gov.md5	464	2013-10-14 08:21:45 +0000	2013-10-14 09:52:23 +0000

Figure 4-7 Dropbox's content of data.dc.gov/2013 and revisions of file data.dc.gov.md5

A hash list of the imaged folder was exported by opening the .E01 file with FTK imager. Fingerprints matched both the files in Table 4.3 and the list produced by CDI (Table 4.4), where the file data.dc.gov.md5 in folder "*root*" is missing because is zero sized. Browsing with CDI and FTK showed that modification timestamps were retained after desktop client upload.

Name	**MD5**	**SHA1**
\$cdi$_metadata.txt	9fa2fcbd0e3e3ab495c865a733b752d0	78f0e59a04581ebbbe4988a46ce8dd54d2bad51f
\2013\$cdi$_metadata.txt	716398474c5d10b2559e1394bd55f56d	7eee567da6ade14bc3cc8c3abff92cefd74521b1
\2013\data.dc.gov.md5	5c87472d77ff352f20469baf4648918a	296a78be66c2c727338abdbbd57606ce35f3b280
\2013\data.dc.gov.md5_cdi_rev_metadata.txt	bb80f6263af2371cc4570288870ca48c	1a8310d06ae378dc64e84814bedca91ee4fa5943
\2013\data.dc.gov_rev_1e14336529.md5	590791fb8e75c6b213b4be36c0753cd0	25096eb28b031e2a9595d0251d716dbc7162518f
\crime_incidents_2011_CSV.csv	7e1854bcb6ebe6267650b49e88c43860	867fba1fb14864aeb4826e4f20a5c5e4fe91704b
\crime_incidents_2012_CSV.csv	688047a9186a8254d3e01c374e47b3d8	55efdc1097451fb1ff332ab4fd248639760040f2
\crime_incidents_2013_CSV.csv	afb6d9a2a2ecbca577586cf41284b775	75716055e36e1ddc5ba7ba92ce47f4c82f2230c9
data.dc.gov.md5_cdi_rev_metadata.txt	dee6bd5169c68fe448c37efd67b55794	b0d5f3683a37a5ac082c502e23b266145f2708d2
\data.dc.gov_rev_1114336529.md5	590791fb8e75c6b213b4be36c0753cd0	25096eb28b031e2a9595d0251d716dbc7162518f
\data.dc.gov.sha1	796b74ebc114b0486550f787d5fe93ed	3c2c784649f21af943230450ed70fea68ec6c421
_nuova cartella\$cdi$_metadata.txt	65d80a2397798bcbc68875f91531b8ee	597712bf2b714f7e29251c74f0ae26ded36a785b
\pass_2008_plain.xml	4666f0b18baf059a5f4acdcef0217e0d	8b0c6cd118665c7ed662ef0d0ae7bb37fcc867f3
\pass_2009_plain.xml	0669a81a8d6b6681dcfb444df3428ab5	aa09ee3ff3708d34559d30db7c65f350e38557dd

\pass_2010_plain.xml	fe35c7d30ad6f2ebcfae50b3bbcc2cf8	ad5b29495376da30e52ba10b17294c984d97ebed
\pass_2011_plain.xml	07d3e48f971bd0f4bb40a7e7c32bbfa7	c2778fbaf497938efe730e8efb9a74dfc947a555

Table 4-4 File hashlist.txt produced by the imaging process

- **Google Drive**: Similar considerations can be made for Google Drive. The imaging process took 5 minutes and 41 seconds to terminate, producing a 9586 KB .E01 file, discovering and downloading 72107684 bytes organized in ten files and 1 subfolder. Differently from Dropbox, Google Drive just keeps track of the revision of *data.dc.gov.md5* in folder "2013" and so there is only one file more than the Windows local folder and no other subfolders where created. The lack of one revision explains why 464 bytes less than Dropbox's storage where found. The imaging process thus created three more files: two to host metadata for directories ("*root*" and "*2013*") plus one for revisions metadata. The hash list produced after opening the .E01 file with FTK imager matched both the files in Table 4.3 and the list produced by CDI. Browsing with CDI and FTK showed again that modification dates and times are kept after desktop upload.

- **Microsoft OneDrive**: Process took 6 minutes and 11 seconds to produce a 9580 KB .E01 image. 72107220 bytes were discovered in 9 files and 1 subfolder, just like the Windows local directory. The missing 464 bytes revision in subfolder "2013" compared to Google Drive accounts for the lesser amount of bytes found. Two more files cdi_metadata.txt were produced during imaging, one located in the root directory and the other in the "2013" subfolder. Again a perfect match of all hash lists confirmed that there were no modifications in file contents. This is also true for modification timestamps which were retained, provided that it is used a parameter named "*client_updated_time*" in the JSON formatted answer of Microsoft servers. Otherwise, the previously considered "*updated_time*" was current as it initially coincides with "*created_time*", the moment the file was uploaded to the cloud platform (server time). This seems much like the "*client_mtime*" field of Dropbox answers, whereas for Google Drive the parameter "*modifiedDate*" was considered.

4.7 Discussion

It is important to compare remote data collection to on-field approach in order to roughly estimate the amount of possible information loss. As discussed, even if examiners had full jurisdiction on provider's premises and obtained legal access, the latter is likely to be unfeasible because it may require a long preparation phase, remarkable system downtime, plenty of resources to make disk copies and the near certainty to gather a vast amount of irrelevant information. Nevertheless, assuming that this is possible for a small data center, it is worth wondering if a post mortem on-site imaging process, which includes all the four phases of ISO/IEC 27037 standard, might entail additional advantages compared to a networked logical acquisition. In the following, it will be therefore presented the case of the Hadoop Distributed File System (HDFS), an architecture suitable for cloud storage (Vittal, 2013) that could be necessary to master during an investigation. Indeed, it would be largely out of scope making a comprehensive coverage and therefore a sample situation will give just an idea of the possible issues, even in the favorable situation of a well-known open source technology. Given the number of possible solutions which underlie today's object stores, it would however be always necessary for a forensic examiner to evaluate each and every situation dispassionately, without feeling overwhelmed in advance by the troubles implied in such an endeavor.

4.7.1 On site acquisition of a HDFS based object store

HDFS (The Apache Software Foundation, 2013)a is a resilient distributed file system shaped after the Google File System (Ghemawatt, Gobioff and Leung, 2003) which is able to scale over thousands of commodity servers. Its architecture is made of a master, called Name Node, which implements file system logic and multiple slaves, known as Data Nodes, which blindly host file content sliced in chunks (also known as blocks) of configurable size. Even in the case of limited setups, making a bit stream copy of all DNs without digging in NN working internals, maybe be affected by the inability of uniquely associate possible interesting data to their owner. In the NN, metadata are kept in a binary file called *fsimage* which records file system structure, for instance which chunk belongs to which file. For performance reasons, *fsimage* is not updated at every write operation. Modifications are saved in an in-memory structure and on a journal file called *edits* which is reconciled at startup and from time to time thereafter. Furthermore, the NN keeps track of the placement of chunks only in the volatile memory and periodically queries DNs to refresh the picture of which node holds which chunk. Against this background, a patient digital evidence specialist well supported by provider's professionals, will probably need custom software tools specifically developed for HDFS forensics and a mixed approach made of live and post mortem activities. A possible protocol of operations follows:

1. in the first place, the file system is secured against every possible modification avenue, but not powered off at first;
2. there should be no need to shut the NN down. HDFS is an open source Java project and data structures are documented. So it is better trying to take a snapshot of the in-memory file system image and blocks-DN association, for instance using *Java Native*

Access library. If this were not possible, after having stopped all HDFS processes, *fsimage* and *edits* are copied and reconciled off line;

3. then it is necessary to analyze the file system to ascertain which files and folders, included the *.Trash* directory in user's home, belong to the suspect under investigation. In case of an off line reconciliation, *Apache's Offline Image Viewer* can be leveraged for dumping *fsimage* raw data in a human readable format. In this way also the names of the blocks which composes the relevant files will be known. In a distributed file system these blocks are regular files stored in DN directory tree.
4. either from the taken memory snapshot or through a dig in the NN log files, if available, it is imperative to find the ip addresses of the DN holding the data blocks. This point and the next are crucial for identification;
5. in a standalone file system, such as Ext3 or NTFS, free hard disk clusters are permanently available for inspection. Conversely, in a distributed scenario there might be no concept of unallocated space. Removing a file corresponds to a deletion of the associated blocks in a DN and an addition of a line in NN log, where block name and DNs get recorded. This is the only clue to possibly recover these blocks from DNs with traditional forensics tools, if disk sectors have not been reallocated. Block recovery can be the real added value compared to remote data collection which does not allow to restore permanently deleted content.
6. at this point DNs have been pointed out and can be shut down so to start the collection phase. The impact on provider's business of this activity is hard to foretell. If the storage subsystem features hardware or software RAID redundancy, which is not needed at all for DNs, removed disks can be just replaced. If the number of nodes taken

offline is limited the replication capabilities of HDFS could still handle the situation as a number of server faults and leave the cluster still operative without other users suffer for any data loss.

7. an acquisition phase follows. To reduce possible system downtime, bit stream images of disks could be performed on site, rather than in a lab.
8. the last step entails preservation: integrity protection codes are calculated at the end of the process and due care in handling and storage of copies is observed to prevent "tampering and spoliation" according to ISO/IEC 27037 lexicon.

So what are the revenues of this painstaking process? Content and metadata of allocated blocks could of course be obtained via remote collection and so trashed items and past revisions. The uncertain benefits could stem from the forensic analysis of DN images. In the first place, an inspection of DN file system tables based on names of removed blocks may lead to recover deleted files. Furthermore, a pattern match search or a carving activity on disk unallocated space may reveal interesting sectors or files. For certain, once this content has been restored, its connection with the suspect under investigation must be crystal clear, because there is no other way to associate it to the user. Table 4.5 resumes pros and cons of on-site forensic acquisition. It can be seen that an on-site approach, if possible, must be very carefully planned in the preparation phase from a costs-benefits standpoint and it is likely to be justified only in investigative cases of extreme importance. There may be occasions however, where an on site acquisition is necessary to achieve an intended result, for instance when the cloud provider is not to be trusted or if content cannot be retrieved remotely as it does not offer an adequate API from a forensic point of view.

Pro	Con
Possibility to recover permanently deleted files	Possibly long preparation phase
Possibility to recover valuable data in unallocated sectors	Probable system downtime
	Remarkable resources to arrange disk copies
	Lengthy copy operations
	Likely need to devise new scripts or software tools
	Uncertain benefits due to the difficulty of finding deleted data and associate it to the suspect

Table 4-5 Pros and Cons of on-site forensic acquisition

5. REPEATABILITY IN A REMOTE ACQUISITION SCENARIO

As discussed in section 4.1, remote acquisition tools like *Cloud Data Imager* can be leveraged to perform technical assessments also during the so called phase of *preliminary investigations* (PI), according to the Italian code of criminal procedure[xlvii]. This is the period when the prosecuting attorney, once acquired a *notitia criminis* , performs any needed action to assess whether it rests on solid grounds so that it requires a crime be prosecuted or ask the judge to drop all charges. According to article 111 of the Italian Constitution which introduced the warranties set forth by the so called "fair trial", proofs take their shape after an adversarial debate, where all relevant parties compete as peers before an unbiased judge[xlviii] and witnesses are directly or cross-examined. However, there may be times where it is necessary to anticipate the proof making process during the PI phase as in the case of unrepeatable investigations concerning evidence whose physical condition or state may change. Indeed, traditional forensic activities may entail a chance of digital evidence modification because of poor compliancy to best practices shown by operators or when the very action of handling or powering on a media cause permanent damages to evidences. However, thanks to their extreme resiliency features, cloud personal storages are always on infrastructures with an availability percentage in excess of 99.9 % over the year and risks of mechanical, thermal or electrical shocks do not apply. In the following we therefore evaluate the implications to repeatability of remote acquisitions performed with tools like CDI in case of personal storages we met in the previous chapter, comparing the differences detected among cloud technologies.

5.1. Unrepeatable technical assessments

In the Italian CCP unrepeatable technical assessments are regulated by article 360 which belongs to the set of activities performed by the prosecutor during the period of the PI. Article 360 states that when the prosecutor needs to perform technical assessments concerning places, people or material whose condition may change, he must give notice to all interested parties of the trial (defendants, victims of the crime and all defense attorneys) of time and place of the appointment of his trusted expert witnesses (people with a specific technical competence in the matter to whom the prosecutor can turn according to article 359 of CCP). He must also inform them about the option to nominate their own consultants who are rightfully allowed to participate to all technical sessions, make comments and vet the correctness of operations. Particularly important for Digital Forensics is also the article 117 of Implementing Provisions of CCP which extends the scope of validity of the aforementioned article 360 to the cases where the assessment itself causes modifications to things, places or people which otherwise would not be liable to change. For example an hard disk , if properly stored in an anti- static bag and in the due environment, is most likely expected to be immutable over a reasonable time frame, but the action of powering it on or the failure to use a proper write blocking device during a bit stream image could impose a permanent damage or anyhow cause a modification of its bit patterns. A partaken procedure guarantees the protection of the interests of all trial parties since the period of PI as some activities, once accomplished, cannot be repeated in the future because objects may be irreparably modified. DE acquisitions may belong to this category as their handling could entail changes which cannot be undone.

5.2. Classification of digital evidence acquisitions

A correct regulatory placement of digital evidence acquisitions is fundamental to evaluate their effects in the penal trial. The recalled article 360 of CCP deals with unrepeatable assessments, but these are just a fraction of the possible technical activities which can be found during the PI period (Fasolin, 2012) with different protection levels for the defendant. Digital evidence specialists can therefore also perform urgent assessments and "*surveys*", typically by sampling material on the crime scene, to avert the risk of evidence dispersion or alteration in accordance to article 354 of CCP. This situation entails weakened safeguards for the suspect compared to article 360 of CCP, as defense attorneys are allowed to attend during operations, but as mere observers and without the right to be notified in advance (article 356 of CCP). However, it must be stressed that the code itself somehow counterbalances this lack by defining that DE acquisitions be performed by police officers, if possible, by immediate and faithful copy on write-once medium of original data, paying attention to its integrity and preservation.

Even less protected for the defendant would be a DE imaging according to article 370 of CCP which enables LE to accomplish investigations with the permission of the public prosecutor and on his behalf. In this case no legal assistance is expected as article 370 does not recall the aforementioned article 356 (Durante and Pagallo, 2012). Furthermore, in addition to cases of urgency, article 348 states that LE, in order to secure all source of evidences can accomplish, autonomously or on behalf of the public prosecutor, "*acts*" or "*operations*" possibly relying on experts which cannot refuse to cooperate. Unfortunately, the code does not give a clear definition of such activities nor sets forth a distinction whatsoever among assessments and surveys (Sottani, 2011) and (Casasole, 2013). However, this gap has been filled by case law and legal doctrine which define the survey an action of mere observation, identification and collection of material that is preliminary to an assessment which conversely entails a thoughtful appraisal of the material and a production of an opinion (Aprile, 2003). The amenability of digital evidence acquisitions to mere surveys has been repeatedly ascertained in statements of the Italian Supreme Court of Cassazione according to which: 1) extracting data from a computer is a merely mechanical operation which can be reproduced indefinitely and hence does not involve unrepeatability profiles[xlix]; 2) copying a file from a seized computer does not entail any evaluation activity from a technical or scientific standpoint[l]. Relying on this school of thought the defendants and their defense counsel are not to be necessarily informed in advance when these operations take place. Entirely different conclusions have been more recently drawn by the legal doctrine according to which every man-computer interaction should happen as an unrepeatable assessment with due warranties for all stakeholders and by means of expert witnesses (Fasolin, 2012) . Tonini believes that digital documents undergo the same general principles applicable to every evidence according to which, when technical activities may alter the assessed objects, it is necessary to previously organize an adversarial debate (Tonini, 2012) as stated by the aforementioned article 117 of Implementing Provisions of CCP. The repeatable or unrepeatable nature of digital evidence acquisitions ultimately appears therefore to be linked to the possibility of alteration of the digital media under observation. So in the case of media that are intrinsically read only (such as CDs or DVDs) or when a *post mortem* acquisition[li] of powerable devices is operated by an expert which implements all due technical safeguards a DE acquisition may be deemed repeatable (Fasolin, 2012). These safeguards consists on leveraging:

- software or hardware write blockers which prevent accidental modification of the media;
- uninterruptible power supplies to avert the risk of power outages during lengthy operations which would result in unpredictable effects on the evidence;
- integrity protection codes or (better) digital signatures and certification authority issued timestamps.

Nevertheless, to avert the risk of invalidating an evidence both because there is a residual chance that technical operations can possibly damage it in some way or because the defendant may later disavow its content, the public prosecutor may ensure that an adversarial debate is anyhow established. This is also in anticipation of the day when seized evidences will be possibly returned to their owners, as it can happen for laptops, smartphones or tablets which are often reclaimed back, and from that time on forever modified. However, not always the organization of the activities strictly obeys the provisions of article 360 when the public attorney decides to rely on experts of criminal police: as operations follows a well-known protocol forged during years of best practices and operators are tasked to only acquire the digital evidence without appraising its content (as it happens for surveys), there might be no need to organize a meeting where an expert witness is appointed, some questions for him to answer are formally put on paper and all parties are invited. It is therefore also possible that forensic expert of LE receive a pretty standard proxy from the prosecutor (under the provision of article 370 CCP) which states that: 1) tools and procedures must ensure that evidence is not changed; 2) there is a perfect match between the source and the copies; 3) all defense counsels must be notified about the time and place planned for operations inception so to invite their trusted expert witnesses. This is an hybrid process that lies halfway between articles 370 and 360, but field experience tells that is a perfectly possible arrangement.

Conversely, a major impact on repeatability may happen when *Live* acquisitions come into play. Indeed, sometimes it is not considered useful to shut down a working equipment like a workstation or server, even in the case of a sudden halt caused by pulling the plug, not to lose possible sources of evidence which may reside in volatile memory like active network connections, active processes, running programs, encryption keys or unsaved documents. This not to mention the possibility that relevant data, for instance related to Internet navigation, be purged after the browser closes or temporary information be wiped by housekeeping scripts triggered by a clean system shutdown. Live forensics needs to cope with systems which cannot be initially or permanently stopped and whose ever changing state may make the assessment truly unrepeatable. Consider the case when the forensic expert needs to perform a memory dump[lii]. System volatile memory changes continuously and possibly unpredictably in response to process creation and termination, network connection establishment and teardown or allocation/deallocation requests issued by running programs. Some degree of alteration is actually introduced by the forensic tool itself when its containing USB drive (as an example) is plugged in and new process is created in memory by the operative system. This means that repeating the experiment consisting in a byte stream acquisition of memory content and calculating the resulting integrity code would lead every time to a different hash value. So in the interest of the forthcoming discussion, it is necessary to wonder, when a technical assessment is not completely repeatable, if all modifications occurred are relevant to classify it as unrepeatable (Fasolin, 2012). For instance, losing timestamp information of files as a result of backup restoration as described in section 4.1, perhaps could not bring to evidence invalidation if what really matters is its content and creation or modification times are not fundamental. In this respect, this action could be deemed *substantially repeatable*, meaning that occurred alterations may be not relevant in the context of the trial as they do not impact on the reliability or sufficiency of the evidence. Cesari reaches the conclusion that when variations do not influence the outcomes of a new following assessment so that its nature and characteristics are preserved, their occurrence is not relevant to classify it as unrepeatable (Cesari, 1999).

5.3. Repeatability in the context of personal cloud storages

We are now entering an almost unexplored territory for which a very few previous contributions exist. A relevant related work concerning repeatability of cloud stored content has been produced by Aterno and Mattiucci, according to which imaging of data is a dynamic activity which need to be considered not repeatable (Aterno and Mattiucci, 2013), even if the authors do not make any distinction among cloud service models.

In section 4.4 we made the consideration that ISO/IEC 27037 standard does not explicitly cover cloud storage services and we assumed to deal with non-interruptible mission critical systems which can be reached only remotely. This could lead someone into thinking that we are facing live systems on which only unrepeatable assessments can be made, but this is not necessarily the case. Indeed, we disregard low level activities such as memory captures as these features are not yet allowed by platform APIs. We are rather interested in folders and objects which may instead be pretty stable in the due conditions. This brings us making the first consideration concerning repeatability of remote acquisitions on cloud stores:

- as discussed in paragraph 4.4.4, if it is impossible to safely exclude that third parties cannot alter the remote content because LE has not exclusive access to suspect's storage area or write permissions cannot be removed by the provider from the account, the remote acquisition is to be classified as unrepeatable and an adversarial debate needs to be organized;
- conversely, when those safeguards are present and recalling the circumstance that cloud personal storages cannot suffer from damages imposed by mechanical, thermal or electrical shocks, remote acquisitions targeted to Dropbox can be deemed repeatable as inner hashes are immutable. Acquisitions targeted to Google Drive and Microsoft OneDrive can be deemed either not repeatable as inner hashes change at every experiments due to changing metadata values or can be deemed substantially repeatable if one considers these metadata (for instance a temporary file download link) irrelevant and unable to invalidate the assessment. More on this later.

In section 4.6.2.2 we determined by means of hash checking that every sample file uploaded in a cloud storage remains unaffected in its content when it is copied back. What deserves to be verified at this point is if there is some variation in objects metadata after several experiments. In other words we need to check if two or more consecutive calls to CDI library function *listFolder* or *listFileRevisions (*see section 4.6.1*)* bring exactly to the same result or some data is altered. It is easily understandable that the latter occurrence would lead to ever changing inner hashes each time an imaging experiment of the same folder is executed. We then organized a very simple scenario in which a folder named *RepeatTest* was created under the root for every CSP leveraging the associated desktop client to keep the content synchronized. This folder contains just one file named *original.txt* filled with a very short text that we modified just to create a revision. Now we are able to capture server side answers when functions *listFolder* or *listFileRevisions* are called (the first when browsing *RepeatTest* folder and the second by right clicking of file *original.txt* to show the revisions). Results are discussed in the following sections for every CSP.

5.3.1 Dropbox

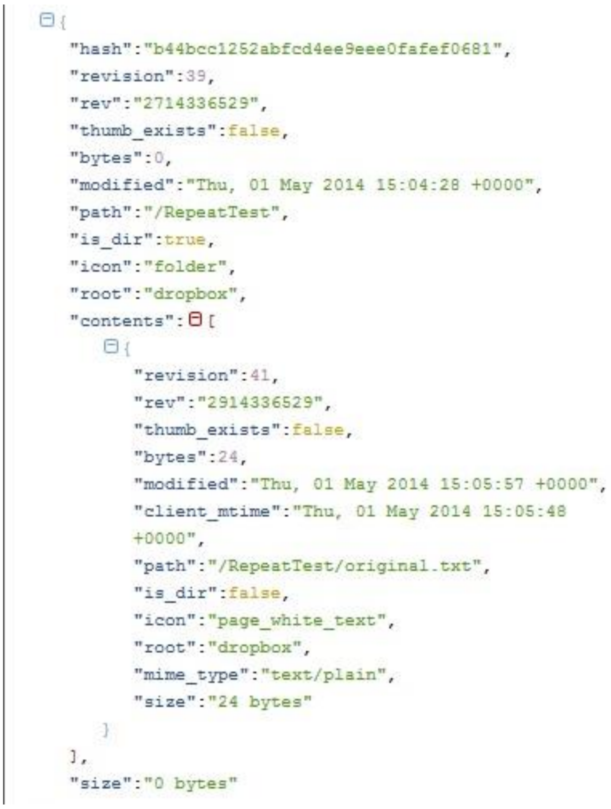
Calling *listFolder* corresponds to a secure HTTP GET containing the keyword */metadata* and folder path in the URL structure. According to Dropbox literature (Dropbox, 2014) the JSON formatted answer contains the fields listed in the following table:

Field	Description
size	A human-readable description of the file size (translated by locale)
bytes	The file size in bytes
path	Returns the canonical path to the file or directory
is_dir	Whether the given entry is a folder or not
is_deleted	Whether the given entry is deleted (only included if deleted files are being returned)
rev	A unique identifier for the current revision of a file. This field is the same rev as elsewhere in the API and can be used to detect changes and avoid conflicts
hash	A folder's hash is useful for indicating changes to the folder's contents in later calls to /metadata. This is roughly the folder equivalent to a file's rev
thumb_exists	True if the file is an image that can be converted to a thumbnail via the /thumbnails call
icon	The name of the icon used to illustrate the file type in Dropbox's icon library
modified	The last time the file was modified on Dropbox, in the standard date format (not included for the root folder)
client_mtime	For files, this is the modification time set by the desktop client when the file was added to Dropbox, in the standard date format. Since this time is not verified (the Dropbox server stores whatever the desktop client sends up), this should only be used for display purposes (such as sorting) and not, for example, to determine if a file has changed or not
root	The root or top-level folder depending on your access level. All paths returned are relative to this root level
revision	A deprecated field that semi-uniquely identifies a file. Use rev instead

Table 5-1 Dropbox metadata field description

Figure 5.1 shows server's answer displayed with JSON Parser Online[liii]. Listing shows details of *RepeatTest* folder which is a child of the root, has size equal to 0, was modified

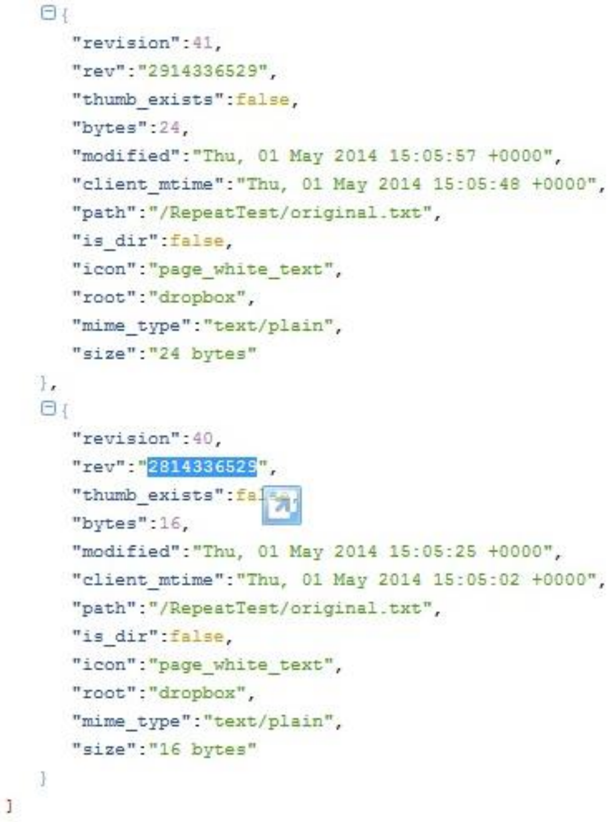
Thu, 01 May 2014 15:04:28 +0000. This folder contains only one file of 24 bytes whose path is /*RepeatTest/original.txt*, was added to Dropbox on Thu, 01 May 2014 15:05:57, was last modified on Thu, 01 May 2014 15:05:48 +0000 and has an revision id equal to 2914336529. Field *is_deleted* is not present as it is only returned for deleted entries.



```
⊟{
   "hash":"b44bcc1252abfcd4ee9eee0fafef0681",
   "revision":39,
   "rev":"2714336529",
   "thumb_exists":false,
   "bytes":0,
   "modified":"Thu, 01 May 2014 15:04:28 +0000",
   "path":"/RepeatTest",
   "is_dir":true,
   "icon":"folder",
   "root":"dropbox",
   "contents":⊟[
      ⊟{
         "revision":41,
         "rev":"2914336529",
         "thumb_exists":false,
         "bytes":24,
         "modified":"Thu, 01 May 2014 15:05:57 +0000",
         "client_mtime":"Thu, 01 May 2014 15:05:48
         +0000",
         "path":"/RepeatTest/original.txt",
         "is_dir":false,
         "icon":"page_white_text",
         "root":"dropbox",
         "mime_type":"text/plain",
         "size":"24 bytes"
      }
   ],
   "size":"0 bytes"
```

Figure 5-1 Dropbox answer to listFolder call

Calling *listFileRevisions* translates into a secure HTTP GET containing the keyword */revisions* and file path in the URL structure. Server's answer contains two entries whose values are already described in table 5.1. As depicted in figure 5.2 the topmost entry corresponds to the most updated version of file *original.txt*, whereas the other is an older version added to Dropbox on Thu, 01 May 2014 15:05:25 +0000, was last modified on Thu, 01 May 2014 15:05:02 +0000 and has a lower revision id equal to 2814336529. It can be seen from figures 5.1 and 5.2 that none of the values corresponding to fields in table 5.1 could change over time and issuing repeatedly a *listFolder* call for *RepeatTest* folder and a *listFileRevisions* for file *original.txt* always gives the same results. We confirmed this fact by repeating the experiment of imaging the *RepeatTest* directory after days and weeks finding that inner hashes were always immutable. We can then conclude that Dropbox's remote acquisitions are **repeatable** from a technical point of view and under the hypothesis of excluding uncontrolled avenues for modifications of data and troubles due to account expiration or provider exiting this kind of business, they can be safely repeated over time getting the same inner hashes. The same holds true for hash lists produced during imaging operations , but not for outer hashes due to changing service data of NTFS file format, such as virtual volume serial number or recording dates.



```
 ⊟{
    "revision":41,
    "rev":"2914336529",
    "thumb_exists":false,
    "bytes":24,
    "modified":"Thu, 01 May 2014 15:05:57 +0000",
    "client_mtime":"Thu, 01 May 2014 15:05:48 +0000",
    "path":"/RepeatTest/original.txt",
    "is_dir":false,
    "icon":"page_white_text",
    "root":"dropbox",
    "mime_type":"text/plain",
    "size":"24 bytes"
 },
 ⊟{
    "revision":40,
    "rev":"2814336529",
    "thumb_exists":fa
    "bytes":16,
    "modified":"Thu, 01 May 2014 15:05:25 +0000",
    "client_mtime":"Thu, 01 May 2014 15:05:02 +0000",
    "path":"/RepeatTest/original.txt",
    "is_dir":false,
    "icon":"page_white_text",
    "root":"dropbox",
    "mime_type":"text/plain",
    "size":"16 bytes"
 }
]
```

Figure 5-2 Dropbox answer to listFileRevisions call

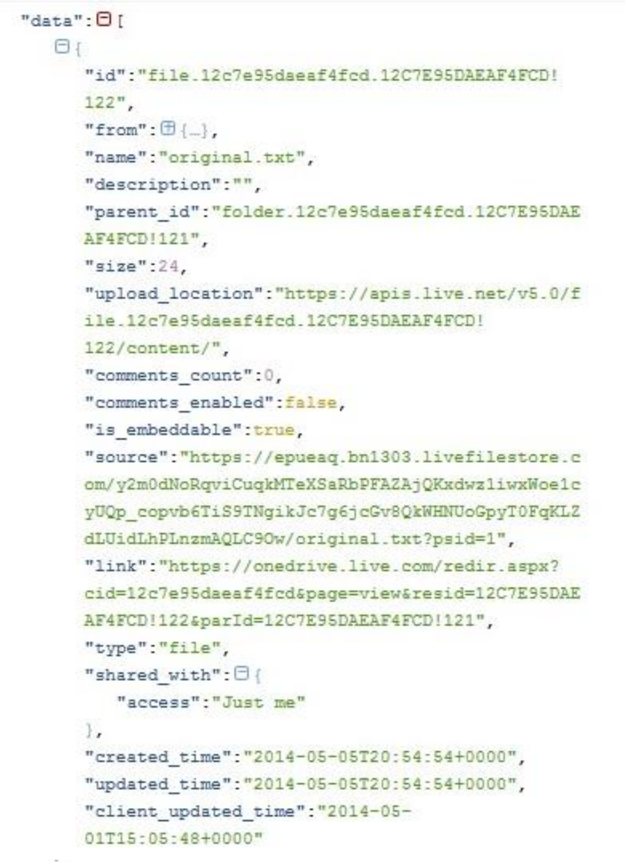
5.3.2 Microsoft OneDrive

Calling *listFolder* corresponds to a secure HTTP GET containing the keyword *me/skydrive/files* for the root or *me/skydrive/shared/files* for shared items or */files* and folder id in the URL structure. According to Microsoft documentation (REST reference Live Connect, 2014) the JSON formatted answer for file and folder objects contains the fields listed in the following table:

Field	Description
id	File or Folder ID
name (from object)	The name of the user who created the folder or uploaded the file
id (from object)	The ID of the user who created the folder or uploaded the file
name	The name of the folder or file
description	A description of the file or folder , or **null** if no description is specified
count	The total number of items in the folder (returned for folders only)
parent_id	The ID of the folder the file or folder is currently stored in
link	The URL of the folder, hosted in OneDrive or a URL to view the item on OneDrive
size	The size, in bytes, of the file (returned for files only)
upload_location	The URL to upload items to the folder hosted in OneDrive or The URL to upload file content hosted in OneDrive
comments_count	The number of comments that are associated with the file (returned for files only)
comments_enabled	A value that indicates whether comments are enabled for the file. If comments can be made, this value is **true**; otherwise, it is **false** (returned for files only)
is_embeddable	A value that indicates whether a file or folder can be embedded. If this folder can be embedded, this value is true; otherwise, it is false.
source	The URL to use to download the file from OneDrive (returned for files only). This value is not persistent.
type	The type of object; "folder" or "file"
created_time	The time, in ISO 8601 format, at which the folder or file was created
updated_time	The time, in ISO 8601 format, that the system updated the file last
client_updated_time	The time, in ISO 8601 format, that the client machine updated the file last
access (shared_with object)	Info about who can access the folder (for example "Just me")
sort_by	Sorts the items to specify the following criteria: updated, name, size, or default

Table 5-2 Microsoft OneDrive metadata field description

Figure 5.3 shows server's answer displayed with JSON Parser Online. Differently from Dropbox, there are no more details about *RepeatTest* folder and the answer consists in an array (called "*data*") of one file objects having size of 24 bytes and ID file.12c7e95daeaf4fcd.12C7E95DAEAF4FCD!122 , which was added to One Drive on May the 5th at 20:54:54 UTC and was last modified by the client application on May 1st at 15:04:48 UTC. As stated in section 4.6.1.4 at the moment there is no way to retrieve data concerning deleted files and their past versions so there is no equivalent of Dropbox's *is_deleted* field in table 5.2 and *listFileRevisions call* is not applicable.



```
"data":[
   {
      "id":"file.12c7e95daeaf4fcd.12C7E95DAEAF4FCD!
      122",
      "from":{…},
      "name":"original.txt",
      "description":"",
      "parent_id":"folder.12c7e95daeaf4fcd.12C7E95DAE
      AF4FCD!121",
      "size":24,
      "upload_location":"https://apis.live.net/v5.0/f
      ile.12c7e95daeaf4fcd.12C7E95DAEAF4FCD!
      122/content/",
      "comments_count":0,
      "comments_enabled":false,
      "is_embeddable":true,
      "source":"https://epueaq.bn1303.livefilestore.c
      om/y2m0dNoRqviCuqkMTeXSaRbPFAZAjQKxdwz1iwxWoe1c
      yUQp_copvb6TiS9TNgikJc7g6jcGv8QkWHNUoGpyT0FqKLZ
      dLUidLhPLnzmAQLC9Ow/original.txt?psid=1",
      "link":"https://onedrive.live.com/redir.aspx?
      cid=12c7e95daeaf4fcd&page=view&resid=12C7E95DAE
      AF4FCD!122&parId=12C7E95DAEAF4FCD!121",
      "type":"file",
      "shared_with":{
         "access":"Just me"
      },
      "created_time":"2014-05-05T20:54:54+0000",
      "updated_time":"2014-05-05T20:54:54+0000",
      "client_updated_time":"2014-05-
      01T15:05:48+0000"
```

Figure 5-3 Microsoft OneDrive answer to listFolder call

It can be seen from table 5.2 that the only value which may change over time is field "***source"*** which is a temporary URL to the file download location. We confirmed this after repeating the call to *listFolder* function which returned every time a different virtual directory name. We then draw the conclusion that OneDrive's remote acquisitions, even under the usual hypothesis of lack of uncontrolled avenues for modifications of data and service availability, could be considered **not repeatable** if one just values the fact that inner hashes will change at every acquisition because of changing values of "*source*" field.

Conversely, it can be judged **substantially repeatable,** if one goes deeper and assumes that these variations do not affect core metadata and will not impact the relevance of acquired data. After all, download link structure is a provider related service information which could appear a trifling detail to the court. Changing metadata at each experiment will impact on the hash list produced during imaging operations which will also change in the part of xxx_metadata.txt files, but hashes relative to objects however will not.

5.3.3 Google Drive

A behavior much similar to Microsoft OneDrive can be detected for Google Drive APIs. Calling *listFolder* corresponds to a secure HTTP GET containing the keyword */files* and folder id in the URL structure. According to Google literature (Google, 2014) all present and deleted files are returned because *trashed* query parameter defaults to true. The JSON formatted answer for file and folder objects contains a plethora of fields that would be too long to describe. It is worth noticing however what metadata change every time a *listFolder* command is issued:

Field	Description
etag	Identifier assigned to the file as per HTTP protocol. It changes as it reflects modification of other retuned fields
thumbnailLink	A temporary link to download file's thumbnail
downloadUrl	A short lived download URL for file content

Table 5-3 Google Drive temporary metadata field description

downloadUrl field validity can be measured in hours so it will not change for closely run remote acquisitions, but tests accomplished after some day reveal that this link do change. Similar considerations apply to *listFileRevisions* calls. Conclusions concerning the repeatability of assessments follow the same reasoning seen for Microsoft OneDrive: strictly speaking they could be considered **not repeatable** because of ever changing inner hashes due to fields in table 5.3. **Substantial repeatability** could however be determined before a court that decide to neglect those changing metadata (which again are CSP service parameters which seem not to bring any further contribution of knowledge and may be deemed irrelevant).

6. THE ALMANEBULA FRAMEWORK

We now completely switch our point view and consider the Cloud not as a target of investigations anymore, but as an ally to forensic investigators which may allow to analyze efficiently huge amounts of digital evidences and information sources, possibly belonging to the category of Big Data, extract actionable knowledge from them and share the results among authorized subjects according to their level of clearance. Modern Forensic Computing, the science that deals with techniques and procedures for identifying, preserving, analyzing and presenting digital data that could be relevant in a court of law (McKemmish, 1999), requires a sharply increasing amount of IT resources as the number of computer related investigations continues to grow. The pervasive presence in a case of electronic devices, always more heterogeneous, connected and capable, forces a forensic expert to manage the availability of gigantic storage areas to host the copies of their memory and the result of their analysis. Furthermore, an efficient strategy that keeps acceptable delivery times, calls for a huge computational power, not only to visualize manifest or hidden content from a single device, but also to extract actionable information from a collection of evidences analyzed as a whole. In this scenario, not only traditional standalone tools may fall short, but also forensic platforms based on a classic three tiered approach (client, application server and central database) may prove themselves inadequate because of their intrinsic inability to scale in and out under the pressure of varying workloads. Classic IT architectures resort to over provisioning to accommodate the demand bursts but, due to the wide difference between peak and average utilization (Armbrust, et al., 2009), their resources may lie pretty undersubscribed. Digital Forensics requires a degree of processing power on large collections of documents which has much in common with the Big Data handling that can offer its technological advances to evidence analysis. It is therefore imperative coping with these technological aspects to stay current to modern scenarios. In this respect, the wealth of available open source toolboxes can considerably help building efficient, cost effective and elastic applications. This chapter delves into a set of design principles, technical specifications and conceptual architecture of a novel forensic platform called *AlmaNebula*, which leverages the power and storage capacity of private/community cloud platforms. A modular petabyte-scalable infrastructure geared towards the automatic extraction of actionable knowledge from a collection of digital evidences exposed by means of intuitive interfaces. This aims to embody the concept of "Forensics as a service", a facility for examiners with very basic technical experience that public or private organizations may utilize to grasp all the benefits offered by the utility computing paradigm.

6.1. The cloud as an harbor for forensics services

Theoretically, it could be admissible to host digital evidences in a public Cloud, if the provider were able to offer Government certified services with proper security category as in the case of the United States FedRAMP. Public infrastructures allow a low time to market, almost limitless computational power or storage, high service availability, disaster tolerance and are often advertised as adhering to severe security and auditing standards. Conversely, sharing control on valuable data unavoidably raises concerns about its availability, confidentiality and legal compliance as the public offer cloud services is not always as transparent as it should be to grant a reasonable peace of mind. As discussed in section 2.4, taking the decision to move to a commercial cloud provider is not only a technical option, but rather a complex management process which aims at correctly identifying the risks and possibly accept and minimize them when balanced by adequate benefits. Central to this risk management plan is contracting some good SLA with marble carved clauses that state provider's accountability for information loss or exposure and a data takeout policy in a well-documented format. At a minimum, the SLA should allow the customer to perform, directly or by means of a trusted third party, a rather complete scrutiny concerning: 1) the relevancy in the customer's country of the security certifications achieved by the CSP; 2) the criteria for selecting, enforcing and monitoring security controls (for instance, it is very important to have insights on aspects like employee lifecycle management or system administration procedures); 3) the localization of the data and possible issues of applicable jurisdiction issues from its migration; 4) compliancy to norms and regulations, in particular concerning data privacy; 5) the business continuity policy. Therefore, the natural conclusion is that, at this stage of maturity of public cloud offers, a framework for evidence analysis is more likely to be targeted towards a private or community cloud deployment. Discussion will not delve further into legal implications and assumes that such a platform is always feasible as, at least in a private deployment with augmented security measures due to resource pooling, court authorizations that were obtained for evidence handling with traditional tools continue to stand.

6.2. Previous and related work

Papers on advances of forensic platforms (Roussev and Richard, 2004) stressed the need of a new class of applications that could harness the power of distributed computing as standalone forensic tools, albeit well designed, could fail to deliver timely results. This happens under the thrust of the massively growing amount of cheap storage at user disposal and the resulting growing request of computational resources needed to handle it. As files extracted from an evidence constitute the most natural atomic unit for a cooperative processing, they proposed a prototypal framework running on a Beowulf class cluster and having a central process that distributes computing tasks to several worker entities and finally aggregates the result. An algorithm able to split an input information into pieces that can be dispatched to many remote computational units and then merge the intermediate artifacts in a final result was later formalized in the MapReduce programming model **MR** (Dean and Ghemawat, 2008). (Roussev, Wang, Richard and Marziale, 2009) acknowledged that MR is a powerful conceptual model for describing typical forensic processing but, at the same time, expressed concerns about the efficiency on small deployments of **Hadoop** [liv], an open source implementation of MR, competitor of the proprietary Google's implementation. This was because of the possible lesser efficiency of Java compared to C and the reduced I/O capacity of the Hadoop File System, built as an abstraction layer on top of regular file systems (Roussev, Wang, Richard and Marziale, 2009). The same authors then devised a framework, named **MMR**, based on the Phoenix shared memory implementation of Map Reduce (Ranger, Raghuraman, Penmetsa and Kozyrakis, 2007) that could scale in cluster environments because inter-node communication is handled by a MPI compliant library (Message Passing Interface Forum, 2009). The **Sleuth Kit** (TSK) (Carrier, 2013)b is an open source library and a collection of command line tools built upon it, that is able to parse the most widespread file system formats (NTFS, FAT, HFS+, Ext2, Ext3, UFS1 and UFS2) packaged in an evidence image file and extract files (whether manifest or possibly deleted) along with metadata and unallocated sectors. TSK constitutes the foundation of many open source forensic tools and platforms that either embed the library in the code or parse the output of the command tools. **The Sleuth Kit Hadoop Framework** (Carrier, 2012) is a very interesting experimental project that relies on TSK and Hadoop to build a distributed system for evidence content extraction, analysis and reporting that is amenable for a cloud deployment. Despite many useful analysis features like text extraction, keyword search and document clustering have been implemented, there is no user interface yet and process outcomes are delivered as JSON report files. The **Open Computer Forensic Architecture** (OCFA) (Vermaas, Simons and Meijer, 2010) is a well-designed forensic platform that was designed having in mind scalability, modularity and openness. It aims to automate content extraction of files from digital evidences and it creates a searchable index of text and metadata that can be queried by mean of a web browser. OCFA is organized in pluggable modules (either derived from already available tools or user created) that recursively process an evidence E under the control of a dispatching entity called the router, which decides what module to invoke next according to the information carried by E. However, module development follows a proprietary schema and persistency of data is delegated to a sound, but monolithic PostgreSQL database. In (Garfinkel S. L., 2010), often cited hereinafter, an outlook of the digital forensics research in the next 10 years is presented, where the author reviews the limitations of today's tools and finds that they are monolithic applications designed to make visible what investigators are loking for, when the mere presence of a file is an evidence of a crime, but fail to detect information that is out of the ordinary or out of place. The need of more intuitive user interfaces able to present information and knowledge to analysts and not only mere data is also covered in (Beebe, 2009).

6.3. Limitations of current approaches

Computer Forensics teams, which typically run understaffed, would appreciate the opportunity to be relieved form daily IT management activity and, more importantly, exploit the potentially vast computational power and storage capacity of Cloud Computing (CC) to harbor and analyze digital data. The Cloud would make it possible to create elastic and available forensic analysis platforms that can grow or be shrunk according to the complexity of the required calculations or the size of the evidences. An infrastructure able to cope with demand peaks with no service disruption and, conversely, no fear of resource wastage during idle times. The same may not hold true for the main currently supported free software or open source solutions that we are going to briefly review in the following:

- **Autopsy**: The Brian Carrier's seasoned forensic browser reached version 2.24 (Carrier, 2013)a and offers a pretty basic way to navigate the directory tree of a disk image, with very useful additional features like keyword searching or file timeline reconstruction. It is now backed up by version 3, which is a java-based complete rewrite with major improvements concerning 1) *Performance*: the tool doesn't parse anymore the outcomes of the TSK-based command line tools of, but rather use the quicker Java Native Interface to call TSK library C functions; 2) *Architecture*: the modular structure will allow an extension of functionality by mean of plug-ins that leverage existing open source tools; 3) *Flexibility*: The result of disk image processing is stored in a SQLite serverless database for faster retrieval at a later time. The new release (currently available for Windows only) relies on identity management services provided by the operative system and it appears conceived for single users running standalone machines.

- **Ptk**: PTK Forensics Basic Edition (Forte, 2008) is an alternative to Autopsy based on a traditional three tiered LAMP architecture (Linux, Apache, Mysql and PhP) and relies on TSK command line tools along with other forensic applications for the heavy lifting of content extraction and analysis. Results are presented to the user by mean of a rather complete web based interface that includes, among other features, a powerful indexing engine. Concurrent case manipulation is possible due to identity management based on Username/password authentication.
- **Dff**: The Digital Forensic Framework (Altheide and Carvey, 2011) is a single user standalone application written in Python and C++ for many operative systems, that features a nice GUI and is pretty extensible thanks to its modular architecture. There are modules for many processing tasks, ranging from file browsing and volatile memory dump analysis to hash comparison and file type statistics.
- **PyFlag**: even if the last release of its source code on Sourceforge dates back to September 2008, because of its forward thinking architecture the Python Forensic and Log Analysis Gui (Cohen, 2008) is still worth a mention. PyFlag is a three tiered framework (backed up by a MySQL database) born to perform computer and network forensics analysis. Its Virtual File System (VFS) constitutes a powerful abstraction where many different source of information like network captures files, log files or disk images files can be unified under as single mount point. A file system loader is in charge of abstracting the real nature of the underlying source. For instance in the case of a *tcpdump* formatted file, all the packets will be reassembled in streams and loaded in the database as objects of the VFS (called *inodes*), which feature an internal ID plus a

string that represents a series of operations (concatenated with a pipe) needed to get to each data. These information is a path for *Scanners*, modules that further processes this data at higher level to produce user consumable information and possibly create new inodes as in the case of file extraction from a zip archive.

The aforementioned frameworks have important conceptual mainstays such as the modular architecture that lets functionalities to be extended or, given the practical lack of standardized formats for file systems metadata representation (Garfinkel S. L., 2010), the possibility to use a database as a central storage for data interchange among disparate modules. However, despite the great added value they bring to the computer forensic community, it is worth noticing the following circumstances:

- **User interfaces made for experts**: there is an important distinction between the role of a forensic expert and the one of an analyst. The former, according to a limited knowledge about the case, prepares the ground for the latter by setting up a container where to put all potentially interesting material, be it manifest or hidden, because of was deleted, concealed or encrypted at the time of evidence seizure. Tools to accomplish this tasks necessarily have complex interfaces in order to allow operations that are close to physical nature of devices and that is why all the listed tools have file system browsing facilities to visualize the directory structure of disk partitions, with advanced features such as enumeration of unallocated sectors or display of file raw content. Nevertheless, this wealth of details is not well suited for analysts, whose aim is uncovering and linking logically hidden information buried in a huge mess of irrelevant data by exploiting their deep acquaintance of the case. Evidence-oriented design of interfaces (Garfinkel S. L., 2010) enable technicians to visualize what they are looking for, but do not much help investigators to extract and consolidate actionable knowledge.

- **Hardware platform scaling**: wherever possible, standalone platforms can improve their performance with *vertical scaling* (Hewitt, 2011), that is empowering the existing hardware by adding more CPUs, disks and memory banks. This can help to temporarily solve the problem of an increased computational and storage peak load, but one is likely time-shifting the moment when a new and more costly monolithic architecture will be needed. Furthermore, this is a rigid and coarse grained method to scale out, so there is some risks of average underutilization during periods of reduced demand.
- **RDBMS issues**: public domain relational databases (RDBMS) such as SQLite, MySQL or PostgreSQL are a natural choice to represent a data model with relationships among entities. RDBMS are rock solid data storages that support a simple, but powerful Structured Query Language to perform operations on records and enforce ACID transactions. These properties are fundamental in all class of real time applications like airline booking or e-commerce which cannot tolerate an inconsistent database status that could be originated, for instance, if two customers accessing the system at the same time were both able to book the last remained seat or the last available item. Conversely, RDBMS may bring some issues as of performance and scalability (Hewitt, 2011) when the amount of information to handle reaches the Web scale: 1) *joins*: well-structured relational models call for schema normalization according to Codd's normal forms and consequent creation of additional tables to manage attributes with rank of autonomous entities and many to many relationships

among entities. At query time, these tables need to be merged together with join operations that are inherently slow; 2) *latency*: when vertical scaling isn't viable (anymore) under the pressure of increasing loads, one can think to approach a distributed RDBMS, where tables are split across several servers. In this scenario, strictly enforcing the ACID paradigm means orchestrating distributed operations where resources are locked waiting for the commit of a previously initiated transaction (Hewitt, 2011). While this can be perfectly acceptable in a high speed local network where wait times are kept small, it could cause long delays when remotely located servers experience outages or because of the latencies of long haul links; 3) *schema*: relational databases call for a precise up front modeling of tables and columns before queries on data can be organized. This approach requires a considerable preliminary design effort as further modifications may directly reflect, possibly at a large extent, on the low level "plumbing" code that interconnects application and database.

- **Basic security**: All the listed forensic tools rely on the authentication services either provided by the operative system or by the application itself. In a cloud scenario however, there is the need to go beyond the baseline security features offered by password based authentication as a cloud targeted framework is likely to be hosted in a multi-tenant environment (Mell and Grance, 2011), where several users may access applications from the public Internet, with a resulting actual risk that a vulnerable virtual machine could become a bridgehead to attack other resources. Cloud platforms entail a remarkable value concentration which may increase the attack surface. It is therefore necessary to strengthen the protection perimeter of information as single factor authentication schemes may no longer suffice to guard sensitive data against all possible security threats.

6.4. Design goals

Based on the previous assumptions, *AlmaNebula's* design rests on the principles stated in the following sub-paragraphs.

6.4.1 Cots driven scalability

A major design goal is achieving an horizontal massive scalability by leveraging commodity off the shelf (COTS) hardware: no special shared redundant storage is requested, but directly-attached hard disks that every server can host internally. The infrastructure should be made of computational units (nodes), possibly arranged in racks and connected to pretty general Ethernet network switches, typically up to 1 Gigabit per second, with a low cost per port. Overall capacity increase should be reached by seamlessly adding new nodes to the network, with no theoretical upper bounds and without service disruption. As far as possible, nodes must be peer, without any specialized role that could become a single point of failure. Elasticity should be possible by mean of automatic facilities that keep under measure machine resources and decide autonomously to intervene when load reaches some upper or lower thresholds. Usage of COTS coupled with an high level of automation will contribute to lower maintenance costs and achieve a relevant degree of investment protection by leveraging existing hardware assets.

6.4.2 Resiliency

The platform should be able to tolerate faults occurring at component level, even when they are so severe to bring down one or more nodes possibly located in different racks. Here the traditional approach of unreliable software based on expensive reliable iron is reversed: fault tolerance and availability is achieved by putting intelligence in the software layer and account for failures that may occur more often to commodity hardware.

6.4.3 Distribution

A geographical distribution of the computational units should be possible, each of which can handle both on line data as well as off-line replicas created for disaster recovery purposes. Data replication protocols are expected to be efficient and resilient enough to cope with temporarily slow or intermittent WAN links. A distributed forensic system is valuable not only to aggregate storage and computing power, but for the possibility to pre-process digital evidences locally and avoid unnecessary transfer of data over costly long haul networks. We can think for instance to a central forensic institution that has some operational branches localized all over the country where digital evidences are available. As it is not always appropriate sending the material with a courier or performing a possibly costly and time consuming bulk network upload of the whole images, one could imagine to extract for example only context related files like documents or access logs and transfer them by mean of a compressed data replication scheme for further processing at the hub.

6.4.4 Parallel processing

As discussed in section 6.2, MapReduce is a conceptual model that cleverly fits to digital evidence processing. According to a publicly available implementation of MR, rapidity of tasks execution should be achieved by leveraging the power of distributed processing. Evidences will be split into atomic entities which will be bestowed concurrently to all online nodes. The boundaries of this entities may vary, but in general they can safely be considered at file level or as chunks in the unallocated space areas of file system. This will parallelize the tasks that can be accomplished independently by each node such as hash calculations.

6.4.5 Loose ACID compliance

As a strict RDBMS ACID compliancy seems not so necessary for forensic applications that are organized in a preliminary write-only batch processing phase where evidence content is extracted and analyzed. As it is usually acceptable that outcomes be available only at the end of the process and that accesses to results made by clients will be mandatorily read-only, we don't expect consistency issues of the database. Therefore, NoSQL technologies that guarantee tunable eventual consistency (Hewitt, 2011) measured in a milliseconds scale could be employed, when a preliminary evaluation foretell benefits in terms of performances, flexibility and scalability compared to RDBMS solutions. Whichever family will be selected, using a database may come very handy in a clustered environment to facilitate modules integration, store files metadata and even content. In this respect, making the server side code DBMS agnostic by means of an abstraction layer trades the performances of a fast, but locking-in native interface for a slower, but portable access method and it could be a wise design choice should one decide to switch from relational to NoSQL databases or vice versa.

6.4.6 Modularity

Some forensics solutions are created with an all-in-one philosophy maybe to simplify training and promote product lock-in (Garfinkel S. L., 2010). Luckily, examples of modular design that leverage third party tools exist in the open source panorama, for instance the already mentioned release 3 of Autopsy or Dff. *AlmaNebula* should be a hosting environment for pluggable modules that, upon registration, will be initialized, executed and terminated according to a user defined pipeline, where the output of one will presumably be the input of the following. Not only will the framework be able to launch modules components, but also to offer some baseline facilities like security, inter-module communication or logging. A module is to be meant like a container of functions which have a predefined common structure, perform related activities and are exposed in a controlled way by means of an interface layer. A module should hide its internals to clients that don't suffer for any change in the code as long as the interface remains stable. Interaction with other modules should find a formal specification in a structured document, usually called *manifest*, where a module presents a list of capabilities such as the functionalities it exports and requires from others along with version level. Structuring *AlmaNebula* as a modular framework would allow to: 1) divide development efforts into smaller parts that can be assigned to a team; 2) reuse existing forensic and information handling tools with minor modifications; 3) realize an incremental path of development. Module development should not be based on proprietary schemes, but rather on well-known solutions, so to attract the widest audience of programmers which could easily reuse their knowledge. In this respect, the OSGi architecture (OSGi Alliance, 2012), a set of specifications that define a dynamic component system for Java, is a notable example, even if the benefits of a modular approach are programming language independent.

6.4.7 Openness

One of the most important restrain factor to a widespread adoption of the Cloud is the fear to be locked into proprietary data formats and technologies as this would have a major impact on many technical and organizational aspects, starting from the possible high costs associated with a provider switch. Instead, a framework which is based from ground up on open standards and possibly on open source would increase the overall level of trust of all parties involved. An open architecture is more easily portable, interoperable (The Testing Standards Working Party), inspectable and subject to contributions. This would bring an important added value, especially to a digital forensics platform that should enable all stakeholders to reproduce all operations in the easiest way. Openness is important also because, in the choice of the underlying cloud platform, portability and interoperability issues should be factored in, as mixed future computation scenarios cannot be excluded a priori. As discussed in section 2.5.1, hybrid cloud deployments are a viable solution when an organization's IT resources are sized to tolerate the average burden, but cannot withstand occasional demand surges. Extra load can then be handled by borrowing computational and storage capacity from an external provider that guarantees and adequate level of performance and trust.

6.5. Requirements

Beyond design principles that inspire the global architecture of *AlmaNebula*, a number of requirements which shape its internals are necessary.

6.5.1 Cloud service model

From the final user's perspective, a Software as a Service model is to be selected. Customers interface will be a web application accessible by any browser or custom apps running on desktop/notebook computers or mobile internet devices. Conforming to the Cloud's philosophy of service programmability, platform features will also be directly exposed, for example by means of SOAP or REST based web services. This would be helpful to allow the final user develop its own interface should the prebuilt application be unsatisfying or make available only a subset of functionalities or maybe in case of an integration with already existing forensic solutions. The backend architecture can be physical or virtualized, even if the latter solution adds a higher degree of consolidation and flexibility in the view of a possible future migration or integration with a third party IaaS. For the sake of portability, virtual infrastructure fabric controller (VIFC), the part of the cloud infrastructure that interacts with VMM to orchestrate virtual machines (VM), should ideally be VMM agnostic or at least implement one that works with well documented VMs file formats and preferably supports VM packaging standards like OVF. The selected cloud ecosystem should expose its compute and storage capabilities via APIs that guarantee the maximum extent of interoperability with other commercial or open source cloud solutions. This should happen natively if possible or by mean of abstraction layers like the Apache *Libcloud*[lv] library, a provider transparent interface for the accomplishment of management tasks such as the creation of VM or object listing in a storage container. Figure 6.1 shows the conceptual service model: by means of a private/public network (N), forensics users (FU) access a cloud application (CA) running in virtual machines (VM) managed by a service provider (SP). These VMs are in turn hosted in an infrastructure, placed on or off SP's premises, under the control of a cloud provider (CP). SP and CP could be different entities or belong to the same organization: no assumptions will be made in this respect, as long as a private/community deployment is enforced, in order to relax the protection mechanism that would be needed by considering a fully public counterpart. In addition to what has been already observed about public clouds, it must be added that some arguments exist against the adoption of PaaS model, albeit this apparently seems perfect to concentrate on development aspects while dropping the burden of IT administration activities. PaaS engines are targeted towards the creation of custom software modules by the general public and therefore usually enforce a strict security model that confine user applications in a sandbox with limited access to OS features and restricted possibilities as to sub process spawning or response times, among others. This may collide with the requirements for the creation and maintenance of an open digital evidence analysis platform that is made of several tools which may need to have low level access to the operative system (OS) functions and leverage the power of any useful DBMS, web server technology or programming language. Furthermore, the risk of locking into proprietary technologies is still remarkable as the standards for application portability such as the "Topology and Orchestration Specification for Cloud Application" (OASIS, 2012) are still draft documents and, even worse, not contributed by cloud founding fathers like Amazon or Microsoft.

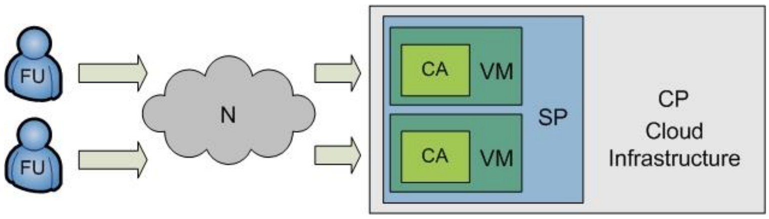


Figure 6-1 AlmaNebula conceptual service model

6.5.2 Alternative analysis

In (Garfinkel S. L., 2010) the author notes that today's forensic tools understandably favor completeness in order not to miss any potentially relevant piece of data. However, there are times where accuracy could be deliberately traded for speed, for instance when it is imperative to achieve a very swift overview of digital evidence content or, maybe, to analyze the same set of evidences with a different software just to timely increase the level of information recall. Following the directory structure, as file based image processing libraries do, translates on many time consuming movements of magnetic HD heads during seek operations. Conversely, processing strategies like stream based disk forensics

(Garfinkel S. L., 2011) efficiently read the evidence material from start to end as a byte stream and extract files by performing a recognition based on known tags or regular expressions. This could be a less complete, but fairly quicker option to analyze a forensic image, especially in presence of unknown or damaged file systems. *AlmaNebula* design therefore requires that practical implementations give the user the opportunity to select when favoring completeness or speed of content and metadata extraction from digital evidences. Even better, this could be considered on a per evidence basis in order to account for evidence storage systems made of modern solid state disks where heads seek penalties do not apply (Garfinkel S. L., 2010).

6.5.3 Information extraction

It is undisputable that evidence processing must start with content and metadata extraction from allocated and unallocated areas of storage devices by means of tools that are either file system structure aware or stream based. Next a shallow analysis phase made of file classification and timeline reconstruction, optional keyword search and document indexing are still valuable practices. On top on traditional information retrieval (IR), that entails a deep domain knowledge as the investigator is required to know in advance what to look for to feed the search engines, it's worth considering an information extraction (IE) layer, where the same data can be viewed from a different unexpected perspective. This is where, without user interaction, named entities like family names, emails or organizations are extracted and linked by means of natural language processing (NLP) algorithms trained on specific corpora or where documents are clustered together according to natural similarities detected using statistical properties of the text they contain (Baeza-Yates and Ribeiro-Neto, 1999). Applying unsupervised IE techniques was found to be helpful in the discovery of events and relations between entities (Louis and Engelbrecht, 2011) and may offer further guidance to investigators because potentially interesting documents somehow autonomously 'pop up' to his attention. Due to its ubiquitous application fields, ranging from business intelligence to brand protection and life sciences, there is a strong interest towards machine processing of texts. Despite developing working IE tools is a very resource consuming endeavor, there are outstanding examples of open source libraries that implement algorithms for data mining tasks like Weka 3[lvi] as well as natural language processing such as LingPipe[lvii], OpenNLP[lviii], GLARF[lix] or the Apache Unstructured Information Management Architecture[lx].

6.5.4 Simplified interfaces

The need to avoid overwhelming the investigator calls for captivating and intuitive interfaces that waive to too technical details of data in favor of knowledge management such as automated link analysis, cross correlation and zooming-in to reduce information overhead (Beebe, 2009). In *AlmaNebula's* dashboard there will be no raw content display, logical partition information or directory browsing with screens bloated with files that do not bring any immediate knowledge contribution. Instead, an alternative approach will consist in presenting the user with baskets belonging to predefined general categories (documents, email messages, chats, multimedia and so on) filled with links to files classified according to magic numbers in headers or footers. Additional containers will reorganize the information base according to its content. For instance documents could be grouped based on statistical similarities in their body into predefined categories (categorization) that are case specific (e.g. finance reports). Other buckets could be filled with named entities detected via NLP algorithms. Inside every container each item could still be displayed in a tabular manner, but with a few detail more than the bare name. If requested, it must be always possible for the user to see relevant file metadata (such as size or timestamps), for instance by mouse hovering, linking each item to its source to verify from which position in the evidence it comes from. When dealing with texts or pictures a very short summary or thumbnail displayed next to the file icon could translate in remarkable time savings with long lists.

6.5.5 Case management

AlmaNebula will present enhanced case management features compared to the missing or basic possibilities offered by the most part of the aforementioned tools:

- some evidence details such as acquisition hashes should be populated automatically by parsing logs, if available, in the most widespread formats (e.g. Access Data FTK Imager). The case itself and every evidence that belongs to it should bring along also its history in terms of multimedia or documental content (e.g. pictures or written reports taken at the time of acquisition);
- as far as possible, in addition to the most common disk image file formats, the platform must be able to deal with the most complete variety of data packages. For example, in presence of network captures, import modules should be able to parse high level protocols, extract relevant stream content (such as web pages or email messages) and metadata (e.g. date/time of start and end or ip addresses);
- a role based case handling policy is to be enforced. System administrators will create users accounts or import them from an existing directory service. Managing a case will then involve the definition of a list of possible operations that will be performed according to rights ranging from the ability to assign permissions and tasks, import

digital evidences into the platform, decide how to process them, query the results and read the reports. For mere guidance, by default the case creator will acquire the *Case Owner* role which will have full rights granted and the same will apply to other people appointed by him. The *Investigator* role will be enabled to decide the data processing criteria as described in the following and have read only access to analysis reports and queries. This last ability will be shared with *Stakeholders*. Roles scope will be a single or multiple cases and further rights could be granted or revoked by case owners.

In figure 6.2 an example of case progression status is illustrated as a state machine that evolves from the *Empty* status just after creation to *Ready* status after all evidences have been processed according to a rule set. Due to the distributed architecture, transitional states such as *Loading* should allow the parallel ingestion of more evidences in the platform. Moreover, adding evidences to a case could entail a trivial copy of the whole image content into the platform or, for the sake of room and bandwidth saving, a more sophisticated identification of some desired content and metadata according to templates (see the next bullet). In the latter case, the extraction phase is anticipated from *Processing* to *Loading*.

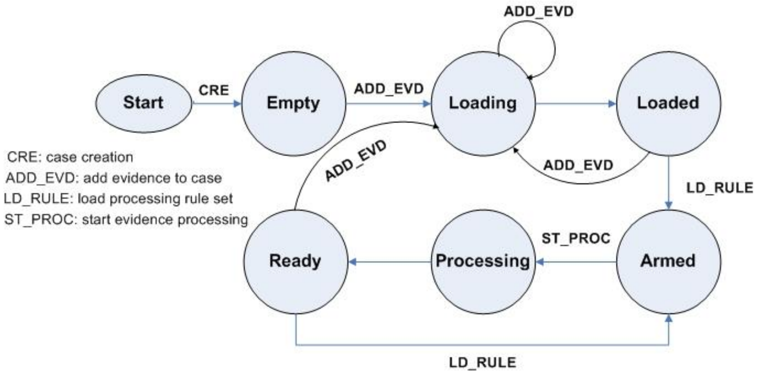
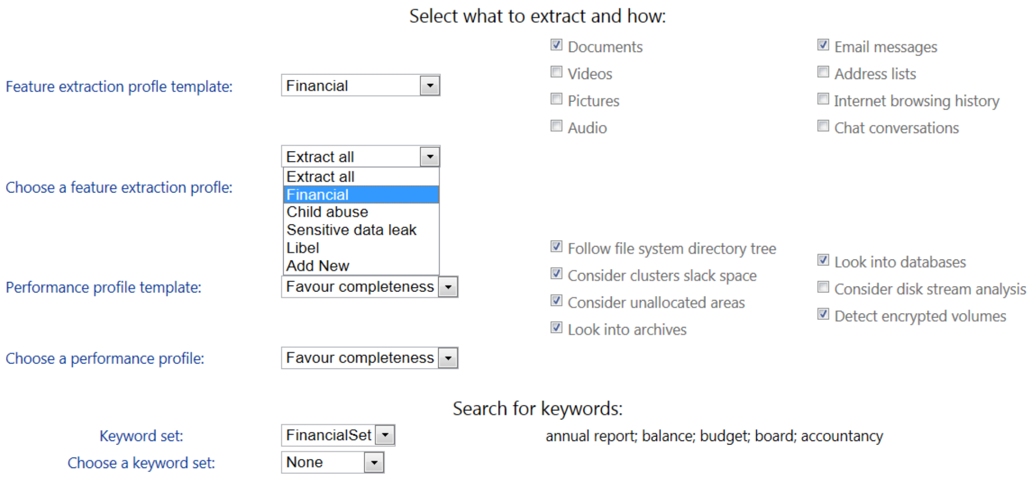


Figure 6-2 Case management state machine

6.5.6 Team collaboration

Enabling multi user case management so more investigators can access a common repository may be not enough to guarantee a fruitful collaboration among all interested parties. Managing a digital investigation is a shared process made of steps in which many different technical, investigative and legal skills are required. Considered the usually different background of all actors involved, collaboration facilities that allow, for instance, for a formal definition of what kind of information must be extracted from the evidences while maintaining an user friendly graphical interface, would be much more productive than statements expressed verbally or written in natural language. An example of evidence processing design is depicted in figure 6.3. An analyst or investigator (and not a forensic expert) is requested to decide what kind of information is worth extracting globally or on a per evidence basis. The interface is as intuitive as possible with prebuilt feature extraction and performance profile templates that could tailor the current case. For instance in a financial investigation could be worth detecting only manifest or deleted documents and email messages instead of Internet browsing history or chat conversations, but with most accuracy by leveraging file systems aware libraries such as TSK that value files metadata too. At other times, just multimedia content could be deemed important to be extracted as quickly as possible with stream tools like *Scalpel*[lxi] or *PhotoRec*[lxii] which carve allocated and unallocated disk areas and cluster slack spaces. In any case, user preferences will be converted to a formal description syntax such as XML or stored in the database for further processing. Secure instant messaging, a wiki for novice members of the team and integration with social tools like Twitter for timely sharing of non-sensitive communications (for example a tweet to announce that the results of the analysis are ready) are just examples of collaboration features that could add an important value during the whole lifetime of a case.



Select what to extract and how:

Feature extraction profile template: Financial

Choose a feature extraction profile: Extract all

☑ Documents
□ Videos
□ Pictures
□ Audio

☑ Email messages
□ Address lists
□ Internet browsing history
□ Chat conversations

Performance profile template: Favour completeness

Choose a performance profile: Favour completeness

☑ Follow file system directory tree
☑ Consider clusters slack space
☑ Consider unallocated areas
☑ Look into archives

☑ Look into databases
□ Consider disk stream analysis
☑ Detect encrypted volumes

Search for keywords:

Keyword set: FinancialSet

Choose a keyword set: None

annual report; balance; budget; board; accountancy

Figure 6-3 An example of evidence processing design

6.5.7 Security

Cloud platforms can be more challenging for digital forensics labs than the usual setups in a private local area network. A ready-to-scale infrastructure must be aware that it could grow and be organized into many geographically dispersed sites possibly communicating over public networks. Furthermore, a major breakthrough would be achieved by granting a secure and ubiquitous network access to mobile users that allowed them to examine all investigation reports via smart devices. Digital evidence can bring a huge added value to a case, so it must be kept safe from prying eyes of external intruders and insiders as it can be easily altered, especially in cloud deployments that are logically siloed, but physically shared. Evidence manipulation could prove even more detrimental for a case than its knowability by unauthorized aliens. In order to avert the risk that it loses its mandatory features of completeness and reliability (Braid, 2001), it is necessary to consider an appropriate level of information assurance as one of the *AlmaNebula*'s design pillars. In the following, a set of minimum security requirements will be specified and practical implementations are free to consider any other measure aimed at enhancing the overall protection level. The threat model considers a private/community deployment and assumes that no harm can come from insiders, notably system administrators, who could observe the state of a VM from the outside (CPU registers, memory space and so on) by means of virtual machine introspection (VMI) (Garfinkel and Rosenblum, 2003). Given the trust relationship with the computing environment, guest VM are assumed globally integer at setup and exposed to risks of cyber attacks only when a connection to a network is operated. Potential victims are both forensic users and cloud applications against which several attacks can be mounted only from outsiders, ranging from theft of credentials to remote exploit of code flaws or misconfigurations, to compromise the integrity and confidentiality of data or get in control of a tenant to attack others. A set of minimum security requirements is specified as follows:

1) **Requirement 1 - Encryption**: data should be protected with strong encryption schemes, preferably based on standard algorithms like AES, when in transit and optionally at rest;
2) **Requirement 2 - User multifactor authentication**: a strong authentication is mandatory for forensic applications irrespective of the type of access. FU must log in by means of a multifactor authentication scheme (Federal Financial Institutions Examination Council, 2005), that is based not only on what user knows, but also on what user has or is. Whichever solution will be selected, it must be considered that in presence of portable appliances like tablets it could more practical to input the authentication code via keyboard instead of plugging smart card or biometric readers. For example, a simple two factor implementation could enforce a traditional

username/password couple backed up by a one-time password (OTP) sent by the platform to the user via SMS. The second factor could also leverage software or hardware tokens compliant to various algorithms designed by the Initiative for Open Authentication[lxiii] (OATH) such as HMAC based One-Time Password (HOTP, RFC 4226), Time based OTP (TOTP, RFC 6238) or OATH challenge response (OCRA, RFC 6287).;

3) **Requirement 3 - Evidence content tampering control**: files content and properties must be hashed and possibly signed upon extraction from disk images or before being imported into the platform so that, if performance penalty is tolerable, every data handling operation can be preceded by genuineness verification to avert the possibility that it was tempered with;

4) **Requirement 4 - Audit trail**: every effect stemming from users interaction with the platform, from login to evidence handling, processing or returned errors must be documented and recorded in a detailed audit log which should be signed, timestamped (if this feasible) and cannot be directly altered through the user interface. An operation log where all steps performed by the platform following user instructions must also be produced. Logs may have more than one verbosity level, should be rotated and kept safe according to corporate security policy;

5) **Requirement 5 - VM monitor (optional)**: Guest VMs should be vetted by integrity monitors like ACPS (Lombardi and Di Pietro, 2011), which leverage VMI to intercept, record and evaluate all suspicious guest activity such as system calls invocation. Monitoring VMs from the VMM allows an effective and hardly detectable way to notice threats like rootkit outbreaks. This must be compared to traditional host based

defense measures which have an excellent view of the system state, but may be detected and subverted by the malware (Garfinkel and Rosenblum, 2003).

6.6. ARCHITECTURE

An overview of the platform proposed as IaaS is sketched in figure 6.4:

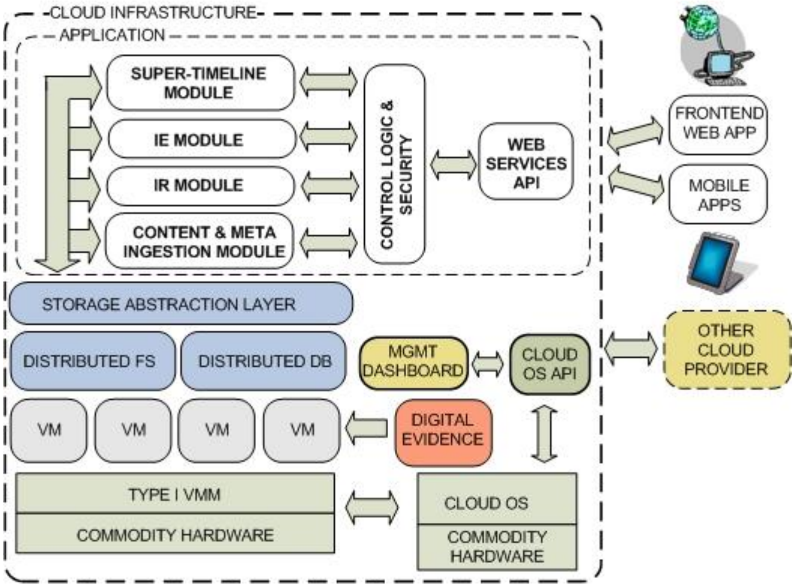


Figure 6-4 AlmaNebula IaaS architecture with type I VMM

6.6.1 Cloud operative system

The Cloud OS plays the fundamental role of fabric controller (VIFC) as it interacts with VMM to manage VMs behavior. It may offer ancillary services such as device block store that can be attached to VM instances (much like USB external drives) or reliable object storing, used for instance for backup purposes. Cloud OS is made of many software components that can run on several commodity machines for load distribution and fault tolerance, but all-in-one deployments should be possible, at least for evaluation purposes. A web services API allows user programs like *Management Dashboards* to monitor and orchestrate the operations of each and every component for example starting, stopping, or metering VMs. The API can be accessed from inside VMs as well in order to consume ancillary services. As already stated, it is desirable that the VIFC be VMM agnostic.

Limiting our scope to open source cloud software solutions, it is worth mentioning OpenNebula[lxiv], OpenStack and CloudStack[lxv] as good candidates for the role of Cloud OS.

6.6.2 Virtual machine manager

A type I VMM setup (Goldberg, 1973) that runs directly on commodity not redundant hardware could be selected, but type II VMMs (hosted) that lie on top of an host OS are also possible. In the latter case, Cloud OS components and VMs can be mixed and matched on the same physical hardware. For portability purposes, VMM should support well documented or, better, standard VM image file packaging like OVF. Type I VMM examples are Xen[lxvi], Microsoft HyperV[lxvii] or VMWare ESX/ESXi [lxviii], whereas KVM[lxix] is a type II solution. Virtual machines host the distributed storage which is created on top of their virtual hard disks. They should be guest OS agnostic. During evidence processing, VMs are started according to availability and the chosen Map function.

6.6.3 Storage layer

An efficient, resilient and distributed storage layer is the foundation of a reliable digital evidence analysis platform whose goal is achieving parallel calculations among peers while scaling seamlessly by adding new nodes to the pool. A database is a convenient way to integrate modules pipelining and store metadata of evidence files in order to leverage filtering and sorting capabilities. File content can be inserted in the database too, even if it could prove more handy using the storage space of the file system, where some existing tools like search engines can process them directly without prior extraction. To cope with this issue, another interesting possibility is realizing a distributed file system as an abstraction layer on top of the database, creating a mediation module that converts POSIX calls such as open() or read() into SQL queries, as the Filesystem in user space project [lxx] (FUSE) shows. Viable DBMS solutions are for example MySQL Cluster [lxxi] in the full ACID compliant domain and Cassandra[lxxii] or HBase in the NoSQL domain, the latter being the DBMS of choice for the Apache Hadoop project. In the storage abstraction layer also the functionalities to shield to applications the internals of DBMS should find their rightful place. If a regular distributed file system is to be preferred, the choice should privilege highly available solutions that could be installed on top of modern journaled file systems like Ext4 or XFS. In this respect, Hadoop native file system HDFS could be an option much like Ceph[lxxiii] or Gluster FS[lxxiv]. The latter are examples of clustered user space file systems that can scale to petabyte and can be a valid HDFS substitute.

6.6.4 Cloud application and API

The cloud application will be made of pluggable modules that will be pipelined under the supervision of the control logic to reflect user configuration. The *Content & Metadata* ingestion module will preliminarily populate the storage and prepare the ground to the following modules, notably: 1) *Information Retrieval* (IR) that will perform text indexing and pattern searching according to exact matching and regular expressions; 2) *Information Extraction* (IE) which will extract named entities and will cluster documents; 3) *Super-timeline* reconstruction that, overtaking the limit of the traditional timeline reconstruction based just on file last modification date/time, will also dig into several log types to rebuild a more comprehensive picture of events. Log2timeline[lxxv] is an outstanding example of an open source super-timeline creation tool that could be repurposed. More modules can be added to perform more functions. A web services based Application Program Interface will expose platform capabilities to web applications and mobile apps after a strong user authentication has been performed by the *Security* module as described earlier in the requirements section.

7. CONCLUSIONS AND FUTURE RESEARCH DIRECTIONS

In this thesis paper we considered the dual role that Cloud Computing platforms can play in relation to Digital Forensics. From the point of view of an investigation, when crime related information is hosted in a cloud storage platform, it may not be possible to follow a traditional approach based on bit stream copying of seized mass memory or rely on cloud provider data delivered without a sound "Forensics as a service". As previous and related work showed, applications devoted to remote data acquisition with forensically sound architectures are not very widespread to date and general purpose tools are used which lack of fundamental features such as read only access or precise audit trails. This opens a broad avenue for the exploration of application program interfaces exposed by personal storage facilities. In this work we demonstrated that, when these interfaces are accessed at the lowest possible level of web services, they are amenable for building valuable forensic tools because of their ability to retrieve existing and trashed files or their past revisions. In this respect, providers are encouraged to empower the capabilities of their programming endpoints by offering functionalities which allow accessing further details such as the ip address of the user workstation and login times. A discussion has been presented concerning the comparison between remote acquisition and on-site collect-acquire approach in the case of the well-known Hadoop Distributed File System, concluding that the latter could be prohibitive, albeit necessary in some occasions to attempt recovering permanently deleted data which would be irreparably lost otherwise. We developed a library which handles write protected access to selected remote folders and masks to overlying applications all the differences existing in several cloud technologies. We also built a prototypal application, namely Cloud Data Imager, which leverages the library to safely browse a remote account and perform a logical copy of all retrievable objects and their metadata in a raw NTFS volume exported to an expert witness container. The first evidences based on stress and functionality tests confirm that CDI faithfully traverses a selected remote directory and more test beds will be performed in the future. Some very interesting development directions include:

- the reliability of the network connection may have heavy impacts on CDI's behavior. Just to make an example, if the network fails while a file is being downloaded the entire process must be restarted as in the aforementioned two failed test runs. Implementing provisions for resuming a download from the point of interruption would certainly contribute to increase the robustness of the application;
- based on the concept of class interface, CDI Library is easily extendable to handle many other storage providers that expose their platforms via http services. Amazon Simple Storage services and Openstack Swift will be first to be evaluated.

Unprecedented results were also achieved in the domain of repeatability of technical assessments concerning cloud personal storages. Field tests have demonstrated that, when external avenues for content modification can be excluded, remote acquisitions accomplished with tools like CDI are always repeatable for Dropbox accounts. For Google Drive and Microsoft OneDrive they can be considered substantially repeatable, if one considers ultimately irrelevant in the context of the trial the ever changing field values as they are internal CSP service information unrelated to the target account core metadata.

Regarding the Cloud as a support tool for clever evidence analysis, the contribution lies in the description of the design goals, requirements and architecture of a novel cloud enabled forensic application which exploits the computational power and storage capacity of collaborating commodity machines to process huge collections of digital evidences. Discussion showed that, compared to some well-established forensic solutions listed in paragraph 6.3, several new features and improvements can be proposed that would be valuable also for non-technical final users. The most important design goals and descending requirements that fuel the platform are resumed in figure 7.1.

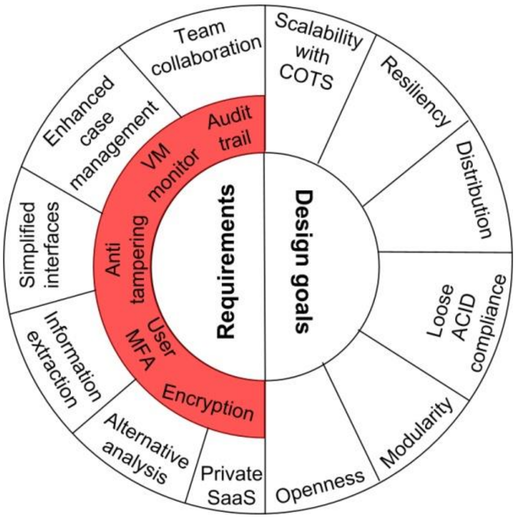


Figura 7-1 Summary of AlmaNebula's design goals and requirements

Next step in the research path is the development of a fully functional prototype to realize the concepts expressed so far. In the testing phase it will be interesting to monitor how evidence processing time varies according to the number of collaborating VMs. Indeed, in presence of a satisfactory Service Level Agreement with the CSP and if legal compliancy to norms and regulations were satisfied, a hybrid or even fully public cloud deployment is to be accounted for as a short/medium term possibility. This could change much the threat model as the hypothesis of trusted computing environment may no longer hold, if malicious activity of insiders are deemed possible, but this much depends on how transparently the cloud provider will be willing to cooperate and share information about its information assurance plan. Adapting the security requirements of outsourced computations to commercial CSP, in a way that also considers the legal issues, will be another major development direction of the current work. Granting an adequate degree of information assurance and regulatory compliance at any time is inherently linked to the level of trust that cloud customers are willing to concede to their CSPs and therefore plays a starring role in the weighted decision of going cloudy. In a public cloud scenario security requirement 5 of section 6.5.7 may not be applicable and, if the hosting environment cannot be trusted in principle, requirement 1 needs to be discussed. Indeed, an abuse of VMI techniques could frustrate state of the art disk encryption if a malicious insider were able to recover decryption keys stored in RAM. In this case, encryption schemes like AESSE (Muller, Dewald and Freiling, 2010), which were devised to resist to cold boot attacks by storing keys in special CPU registers, will fall short if VMI tools have access to those registers. Other approaches that leverage virtual Trusted Platform Modules (vTPM) (Berger, et al., 2006), so that secure storage and cryptographic functions of TPMs[lxxvi] are available to applications running on VMs, are questionable too if vTPMs happen to be under CSP exclusive control. Furthermore, garbage collectors and user programs may not clear memory spaces after deallocation, leaving sensitive content readable by means of VMI access to those spaces. An implementation based on Secure Coprocessors[lxxvii] (Sadeghi, Schneider and Winandy, 2010), where not only crypto keys are stored, but also complex calculations may take place, is interesting, but may collide with the possible lack of specialized hardware in commodity servers and might be feasible only as extra privacy service offered by the provider. Accessing kernel and userland memory from the VMM is to be avoided or at minimum evaded in presence of a possible hostile computational environment, but this much depends on VMM implementation and introspection techniques. Some proposed evasion proof of concepts exploit the fact that VMI libraries like XenAccess, now evolved into vmitools[lxxviii] project, rely on guest operating system kernel integrity to fill the so called semantic gap (Payne, Carbone and Lee, 2007). This consists into mapping the raw view offered by VMI memory page reads into a meaningful high level representation of processes and files. This paves the way to evasion techniques such as DKSM (Bahram, et al., 2010), that are able to present any desired external representation of the VM state by tampering with kernel data structures (syntax or semantics based manipulation).

REFERENCES

Allen, B., Juillet, L., Miles, M., Paquet, G., Roy, J., Wilkins, K. (2004). The Organisational Culture of Digital Government: Technology, Accountability & Shared Governance. In A. Pavlichev, G. D. Garson, *Digital Government: Principles and Best Practices* (pp. 78-96). IGI Global.

Allison, D. S., Capretz, M. A. (2011). Furthering the Growth of Cloud Computing by Providing Privacy as a Service. *Lecture Notes in Computer Science, Volume 6868, Information and Communication on Technology for the Fight against Global Warming*, pp. 64-78.

Altheide, C., Carvey, H. (2011). *Digital Forensics with Open Source Tools.* Elsevier.

Aprile, E. (2003). Le indagini tecnico-scientifiche: problematiche giuridiche sulla formazione della prova penale (Technical and scientific investigations: legal issues of proof making). *Cassazione penale*, 4034-4042.

Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R. H., Konwinski, A., Lee, G. (2009, 02 10). *Above the Clouds: A Berkeley View of Cloud Computing.* Retrieved 02 2012, from http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.pdf

Aterno, S., Mattiucci, M. (2013). Cloud Forensics e nuove frontiere delle indagini informatiche nel processo penale. *Archivio Penale*, 865-878.

Baeza-Yates, R., Ribeiro-Neto, B. (1999). *Modern Information Retrieval.* Addison Wesley.

Bahram, S. B., Jiang, X., Wang, Z., Grace, M., Li, J., Srinivasan, D., . . . Xu, D. (2010). *DKSM: Subverting Virtual Machine Introspection for Fun and Profit.* Retrieved 06 2012, from http://www4.ncsu.edu/~zwang15/files/srds10.pdf

Barroso, L. A., Holzle, U. (2009). *The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines.* Morgan Claypool.

Beebe, N. (2009). Digital Forensic Research: The Good, the Bad and the Unaddressed. *Advances in Digital Forensics V IFIP Advances in Information and Communication Technology* (pp. 17-36). Springer.

Berger, S., Caceres, R., Goldman, K. A., Perez, R., Sailer, R., Van Doorn, L. (2006). *vTPM: Virtualizing the Trusted Platform Module RC23879.* IBM.

Bias, R. (2010). *Elasticity is NOT #Cloud Computing … Just Ask Google.* Retrieved 02 2012, from http://cloudscaling.com/blog/cloud-computing/elasticity-is-not-cloud-computing-just-ask-google

Böhm, A. (2013). *The SWOT Analysis.* GRIN Verlag.

Braid, M. (2001). *Collecting Electronic Evidence After a System Compromise.* Retrieved 05 2012, from http://www.auscert.org.au/render.html?it=2247

Carrier, B. (2012). *Sleuthkit Hadoop Framework Wiki.* Retrieved 02 2012, from https://github.com/sleuthkit/hadoop_framework/wiki

Carrier, B. (2013). *Autopsy Forensic Browser*. Retrieved from http://www.sleuthkit.org/autopsy/index.php

Carrier, B. (2013). *The Sleuth Kit*. Retrieved from http://www.sleuthkit.org/sleuthkit/

Casasole, F. (2013). *Le indagini scientifiche nel processo penale.* Dike Giuridica.

Cesari, C. (1999). *L'irripetibilità sopravvenuta degli atti di indagine.* Giuffrè.

Chetal, A., Ramamoorthy, B., Peterson, J., Wallace, J., Drgon, M., Bhavsar, T. (2011, 07 17). *Interoperability and Portability.* Retrieved 04 2014, from https://cloudsecurityalliance.org/wp-content/uploads/2011/08/Domain-6-First-Draft-IP.docx

Chung, H., Park, J., Lee, S., Kang, C. (2012, 11). Digital forensic investigation of cloud storage services. *Digital investigation, 9*(2), 81-95. doi:http://dx.doi.org/10.1016/j.diin.2012.05.015

Cohen, M. (2008). PyFlag – An advanced network forensic framework. *Digital Forensic Research Workshop V* (pp. S 1 1 2 – S 1 2 0). Elsevier.

Dean, J., Ghemawat, S. (2008). MapReduce: Simplified data processing on large clusters. *Communications of the ACM - 50th anniversary issue: 1958 - 2008 vol 51 Issue 1*, pp. 107-113.

Distributed Management Task Force. (2010). *Open Virtualization Format Specification.* Retrieved from http://www.dmtf.org/sites/default/files/standards/documents/DSP0243_1.1.0.pdf

Dropbox. (2014). *Core API - endpoint reference - Dropbox*. Retrieved 09 2013, from dropbox.com: https://www.dropbox.com/developers/core/docs

Durante, M., Pagallo, U. (2012). *Manuale di informatica giuridica e diritto delle nuove tecnologie.* Utet Giuridica.

Dykstra, J., Sherman, A. T. (2012). Acquiring forensic evidence from infrastructure-as-a-service cloud computing: Exploring and evaluating tools, trust, and techniques. *Digital Investigation, 9 Supplement*, S90-S98. doi:http://dx.doi.org/10.1016/j.diin.2012.05.001

Edwards, D. (2010). *What is DevOps?* Retrieved 03 2014, from http://dev2ops.org/2010/02/what-is-devops/

Effortless .Net Encryption. (2012). Retrieved 10 2013, from https://effortlessencryption.codeplex.com/

Etro, F. (2011). The Economics of Cloud Computing. *The IUP Journal of Managerial Economics Vol. IX, No. 2*, 1-16.

Fasolin, S. (2012). La copia di dati informatici nel quadro delle categorie processuali (The copy of digital data in the context of penal trial cathegories). *Diritto Penale e Processo*, 372.

Federal Financial Institutions Examination Council. (2005). *Authentication in an Internet Banking Environment.* Retrieved 03 2012, from http://www.ffiec.gov/pdf/authentication_guidance.pdf

Forte, D. (2008). The PTK: an alternative advanced interface for the sleuth kit. *Network Security Issue 4*, pp. 10-13.

Garfinkel, S. L. (2010). Digital forensics research: the next 10 years. *Digital Forensic Research Workshop* (pp. S64–S73). Elsevier.

Garfinkel, S. L. (2011). *Digital media triage with stream-based forensics and bulk extractor.* Retrieved 03 2012, from http://simson.net/clips/academic/2011.BulkExtractor.pdf

Garfinkel, T., Rosenblum, M. (2003). *A Virtual Machine Introspection Based Architecture for Intrusion Detection.* Retrieved 05 2012, from http://www.stanford.edu/~talg/papers/VMI/vmi-ndss03.pdf

Ghemawatt, S., Gobioff, H., Leung, S.-T. (2003). The Google file system. *SOSP '03 Proceedings of the nineteenth ACM symposium on Operating systems principles* (pp. 29-43). New York: ACM.

Goldberg, R. P. (1973). *Architectural principles fro virtual computer systems.* Retrieved 06 2012, from http://www.dtic.mil/cgi-bin/GetTRDoc?AD=AD772809&Location=U2&doc=GetTRDoc.pdf

Google. (2014). *About file versions – drive help*. Retrieved from https://support.google.com/drive/answer/2409045?hl=en

Google. (2014). *API Reference - Google Drive SDK*. Retrieved 05 2014, from https://developers.google.com/drive/v2/reference/

Google. (2014). *Concepts and techniques – Google cloud storage – Google developers*. Retrieved 04 2014, from https://developers.google.com/storage/docs/concepts-techniques

Hale, J. S. (2013). Amazon Cloud Drive forensic analysis. *Digital Investigation, 10*(3), 259–265. doi:http://dx.doi.org/10.1016/j.diin.2013.04.006

Hardt, D. (Ed.). (2012). *RFC 6749 - The OAuth 2.0 Authorization Framework.* Retrieved 09 2013, from tools.ietf.org: http://tools.ietf.org/html/rfc6749

Hewitt, E. (2011). *Cassandra The definitive guide.* O'Reilly.

International Organization for Standardization. (2012). ISO/IEC 27037:2012 Information technology -- Security techniques -- Guidelines for identification, collection, acquisition and preservation of digital evidence. Geneva, Switzerland. Retrieved from http://www.iso.org/iso/catalogue_detail?csnumber=44381

Kundra, V. (2010). *25 Point Implementation Plan To Reform Federal Information Technology Management.* Retrieved 02 2014, from https://cio.gov/wp-content/uploads/downloads/2012/09/25-Point-Implementation-Plan-to-Reform-Federal-IT.pdf

Kundra, V. (2010). *State of Public Sector Cloud Computing.* Retrieved from https://cio.gov/wp-content/uploads/downloads/2012/09/StateOfCloudComputingReport-FINAL.pdf

Kundra, V. (2011). *Federal Cloud Computing Strategy.* Retrieved from https://www.dhs.gov/sites/default/files/publications/digital-strategy/federal-cloud-computing-strategy.pdf

Lombardi, F., Di Pietro, R. (2011). Secure virtualization for cloud computing. *Journal of Network and Computer Applications Volume 34 Issue 4*, pp. 1113–1122.

Louis, A., Engelbrecht, A. (2011). Unsupervised discovery of relations for analysis of textual data. *Digital Investigation Vol. 7 Issues 3-4*, pp. 154-171.

Mather, T., Kumaraswamy, S., Latif, S. (2009). *Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance.* O'Reilly.

McKemmish, R. (1999). What is Forensic Computing. *Trends and Issues in Crime and Criminal Justice (n. 118).*

Mell, P., Grance, T. (2011). *The NIST Definition of Cloud Computing NIST Special Publication 800-145.* Retrieved 02 2014, from http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf

Message Passing Interface Forum. (2009). *MPI: A Message-Passing Interface standard Verison 2.2.* Retrieved 02 2012, from http://www.mpi-forum.org/docs/mpi-2.2/mpi22-report.pdf

Metz, J. (2013). *Libewf and tooling to access the Expert Witness Compression Format (EWF).* Retrieved 10 2013, from http://code.google.com/p/libewf/

Microsoft. (2013). *Specifications for the .VHD format for Virtual Hard Disks* . Retrieved 10 2013, from http://www.microsoft.com/en-us/download/details.aspx?id=23850

Milne, K. (2010). *IT Value Transformation Road Map.* Retrieved 4 2014, from http://www.vmware.com/files/pdf/ITPI-cloud-strategy-brief-IT-value-transformation.pdf

Misra, S. C., Mondal, A. (2010). Identification of a company's suitability for the adoption of cloud computing and modelling its corresponding Return on Investment. *Mathematical and Computer Modelling*. doi:doi:10.1016/j.mcm.2010.03.037

Muller, T., Dewald, A., Freiling, F. C. (2010). AESSE: a cold-boot resistant implementation of AES. *EUROSEC '10 Proceedings of the Third European Workshop on System Security* (pp. 42-47). New York: ACM.

Nelson Smith, S. (2010). *What Is This Devops Thing, Anyway?* Retrieved 03 2014, from http://www.jedi.be/blog/2010/02/12/what-is-this-devops-thing-anyway/

Newton-King, J. (2013). *Json.NET*. Retrieved from http://james.newtonking.com/json

OASIS. (2012). *Topology and Orchestration Specification for Cloud Applications Ver 1.0.* Retrieved 04 2012, from http://docs.oasis-open.org/tosca/TOSCA/v1.0/TOSCA-v1.0.pdf

Openstack. (2013). *Chapter 5. Object Storage - OpenStack Cloud Administrator Guide - havana*. Retrieved 04 2014, from http://docs.openstack.org/admin-guide-cloud/content/ch_admin-openstack-object-storage.html

OSGi Alliance. (2012). *The OSGi Architecture.* Retrieved 05 2012, from http://www.osgi.org/About/WhatIsOSGi

Paquette, S., Jaeger, P. T., Wilson, S. C. (2010). Identifying the security risks associated with governmental use of cloud computing. *Government Information Quarterly, 27*(3), pp. 245–253. doi:http://dx.doi.org/10.1016/j.giq.2010.01.002

Parkhill, D. (1966). *The challenge of the computer utility.* Addison-Wesley.

Payne, B. D., Carbone, M., Lee, W. (2007). *Secure and Flexible Monitoring of Virtual Machines.* Retrieved 06 2012, from http://www.acsac.org/2007/papers/138.pdf

Quick, D., Choo, K.-K. R. (2013). Dropbox analysis: Data remnants on user machines. *Digital Investigation, 10*(1), 3-18. doi:http://dx.doi.org/10.1016/j.diin.2013.02.003

Quick, D., Choo, K.-K. R. (2013). Forensic collection of cloud storage data: Does the act of collection result in changes to the data or its metadata? *Digital Investigation*. doi:http://dx.doi.org/10.1016/j.diin.2013.07.001

Quick, D., Choo, R. (2013). Digital droplets: Microsoft SkyDrive forensic data remnants. *Future Generation Computer Systems, 29*(6), 1378–1394. doi:http://dx.doi.org/10.1016/j.future.2013.02.001

Ranger, C., Raghuraman, R., Penmetsa, A., Kozyrakis, C. (2007). Evaluating MapReduce for Multi-core and Multiprocessor Systems. *Proceedings of the 13th Intl. Symposium on High-Performance Computer Architecture (HPCA)*, (pp. 13-24). Phoenix, AZ.

REST reference Live Connect. (2014). Retrieved 09 2013, from msdn.microsoft.com: http://msdn.microsoft.com/en-us/library/live/hh243648.aspx

Roussev, V., Richard, G. (2004). Breaking the performance wall: the case for distributed digital forensics. *Proceedings of the 2004 Digital Forensics Research Workshop.* DFRWS.

Roussev, V., Wang, L., Richard, G., Marziale, L. (2009). A Cloud Computing Platform for Large-Scale Forensic Computing. In *Advances in Digital Forensics V - IFIP Advances in Information and Communication Technology* (pp. 201-214). Boston: Springer.

Sadeghi, A.-R., Schneider, T., Winandy, M. (2010). Token-Based Cloud Computing. *TRUST 2010* (pp. 417-429). Berlin: Springer-Verlag.

Shahine, O. (2012). *New SkyDrive recycle bin available today and Excel surveys coming soon | OneDrive Blog*. Retrieved 04 2014, from http://blog.onedrive.com/new-skydrive-recycle-bin-available-today-and-excel-surveys-coming-soon/

Sottani, S. (2011). Rilievi e accertamenti sulla scena del crimine. *Archivio Penale*, 777-784.

Spyridopoulos, T., Katos, V. (2011). Towards a Forensically Ready Cloud Storage Service. *Proceedings of the Sixth International Workshop on Digital Forensics & Incident Analysis (WDFIA).*

Strauch, C. (2011). *NoSQL Databases.* Retrieved 02 2012, from http://www.christof-strauch.de/nosqldbs.pdf

The Apache Software Foundation. (2013). *Apache Hadoop*. Retrieved 10 2013, from http://hadoop.apache.org/

The Apache Software Foundation. (2013). *Apache log4net*. Retrieved 09 2013, from http://logging.apache.org/log4net/

The European Network and Information Security Agency. (2009). *Cloud Computing Benefits, risks and recommendations for information security.* Retrieved 04 2014, from http://www.enisa.europa.eu/act/rm/files/deliverables/cloud-computing-risk-assessment/at_download/fullReport

The European Network and Information Security Agency. (2011). *Security & Resilience in Governmental Clouds Making an informed decision.* Retrieved 04 2014, from http://www.enisa.europa.eu/act/rm/emerging-and-future-risk/deliverables/security-and-resilience-in-governmental-clouds/at_download/fullReport

The National Institute of Standards and Technology. (2010). *Guide for Applying the Risk Management Framework to Federal Information Systems.* Retrieved 04 2014, from http://csrc.nist.gov/publications/nistpubs/800-37-rev1/sp800-37-rev1-final.pdf

The Testing Standards Working Party. (n.d.). *Integration, Interoperability, Compatibility and Portability.* Retrieved 04 2012, from http://www.testingstandards.co.uk/interop_et_al.htm

Tonini, P. (2012). Il documento informatico: problematiche civilistiche e penalistiche a confronto. *Corriere Giuridico*, 432-439.

Vermaas, O., Simons, J., Meijer, R. (2010). Open Computer Forensic Architecture: a way to process terabytes of forensic disk images. In S. Zanero, E. Huebner, *Open source software for digital forensics.* Springer.

Vittal, C. (2013, 02). *Scalable Object Storage with Apache CloudStack and Apache Hadoop.* Retrieved 10 2013, from http://archive.apachecon.com/na2013/presentations/26-Tuesday/Cloud_Crowd/Chiradeep%20Vittal%20-%20Scalable%20Object%20Storage%20with%20Apache%20CloudStack%20and%20Apache%20Hadoop/S3_HDFS_apachecon.pdf

INDEX OF TABLES

INDEX OF FIGURES

AKNOWLEDGMENTS

A deep sense of gratitude goes to my tutor Prof. Cesare Maioli and to my friend and co-tutor Prof. Roberto Di Pietro for their invaluable support and guidance throughout the course of my PhD.

ENDNOTES

[i] Atomicity Consistency Isolation Durability are the key features of relational databases

[ii] In traditional forensic lexicon, making the dd of a device means performing a bit stream copy of its content to a clone memory support

[iii] This is somehow compatible with the Nielsen's empirical law (http://www.useit.com/alertbox/980405.html) which states that user's Internet connection speed increases by 50% every year. So that a connection having speed S this year will have a speed of S*(1,5)^n in n years

[iv] Representational State Transfer

[v] Simple Object Access Protocol

[vi] Hypertext Transfer Protocol

[vii] Secure Hypertext Transfer Protocol

[viii] http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-regions-availability-zones.html

[ix] http://aws.amazon.com/autoscaling

[x] VMM is the layer of software that virtualizes the physical resources such as CPU, memory and disks to guest operating systems which "think" to be working on physical hardware. Type 1 VMMs (bare metal) run directly on hardware whereas Type II (hosted) are applications run by an underlying operative system

[xi] One standard rack unit corresponds to 1.75 inches or 44.45 mm. Server height is usually expressed in rack units

[xii] http://dev2ops.org/about

[xiii] http://calculator.s3.amazonaws.com/index.html

[xiv] http://www.vmware.com/solutions/consolidation/consolidate.html

[xv] http://csrc.nist.gov/drivers/documents/FISMA-final.pdf

[xvi] One example: http://aws.amazon.com/about-aws/whats-new/2011/09/15/aws-fisma-moderate/

[xvii] https://support.google.com/accounts/answer/3024190

[xviii] http://www.europarl.europa.eu/sides/getDoc.do?pubRef=-//EP//TEXT+WQ+E-2011-006901+0+DOC+XML+V0//EN

[xix] http://www.disa.mil/Services/Enterprise-Services/Infrastructure/RACE

[xx] VMM is the layer of software that virtualizes the physical resources such as CPU, memory and disks to guest operating systems. Type 1 VMMs (bare metal) run directly on hardware whereas Type II (hosted) are applications run by an underlying operative system

[xxi] https://standards.ieee.org/develop/project/2301.html

[xxii] https://standards.ieee.org/develop/project/2302.html

[xxiii] http://europa.eu/rapid/pressReleasesAction.do?reference=SPEECH/11/50

[xxiv] EU's flagship initiative for a flourishing digital economy by 2020

[xxv] Reding's speech at a meeting of the European Privacy Platform Group in Brussels, march 2011

[xxvi] http://ec.europa.eu/information_society/activities/cloudcomputing/docs/ccconsultationfinalreport.pdf

[xxvii] http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=COM:2012:0529:FIN:EN:PDF

[xxviii] https://www.gov.uk/government/publications/government-cloud-strategy

[xxix] https://www.gov.uk/government/publications/uk-government-ict-strategy-resources

[xxx] http://govstore.service.gov.uk/cloudstore/

[xxxi] http://gcloud.civilservice.gov.uk/files/2012/05/G-Cloud-Services-IA-Requirements-and-Guidance-version-1-0-_for-publication_1-2-1.pdf

[xxxii] https://www.cesg.gov.uk/Pages/homepage.aspx

[xxxiii] http://gcloud.civilservice.gov.uk/about/sales-information/

[xxxiv] February 2014 data file G-Cloud-Total-Spend-12-03-14-for-publication.csv last updated 20 march 2014

[xxxv] http://coud.cio.gov

[xxxvi] http://cloud.cio.gov/fedramp

[xxxvii] http://www.finance.gov.au/files/2012/04/final_cloud_computing_strategy_version_1.pdf

[xxxviii] NBN is a high speed fiber to the premises network that is expected to finally reach 93% of Australian territory. The remaining 7% will be connected with wireless and satellite technologies equipment

[xxxix] http://www.ag.gov.au/NationalSecurity/ProtectiveSecurityPolicyFramework/Pages/default.aspx

[xl] http://www.asd.gov.au/infosec/ism/

[xli] http://fiddler2.com

[xlii] https://www.f-response.com/

[xliii] http:/aws.amazon.com/s3

[xliv] http://www.openstack.org

[xlv] http://www.accessdata.com/support/product-downloads

[xlvi] http://code.kliu.org/hashcheck/

[xlvii] art. 326 and following of ccp

[xlviii] Art.1 of Constitutional law 23 nov 1999, n. 2 which modified art. 111 of Italian Constitution

[xlix] Cassazione Sez. I 30 aprile 2009 Corvino in CED Cass. n. 244454

[l] Cassazione Sez. I 5 marzo 2009 Aversano Stabile n. 14511

[li] Post mortem acquisition of digital evidence imply their preliminary shutdown and removal from their location

[lii] Examples of tools for memory capture for Windows are FTK Imager or Belkasoft Live RAm Capturer

[liii] http://json.parser.online.fr/

[liv] Hadoop and its underlying file system HDFS are projects of the Apache foundation (http://hadoop.apache.org/)

[lv] http://libcloud.apache.org/

[lvi] http://www.cs.waikato.ac.nz/ml/weka/

[lvii] http://alias-i.com/lingpipe/

[lviii] http://opennlp.apache.org/

[lix] http://nlp.cs.nyu.edu/meyers/GLARF.html

[lx] http://uima.apache.org/index.html

[lxi] http://www.digitalforensicssolutions.com/Scalpel/

[lxii] http://www.cgsecurity.org/wiki/PhotoRec

[lxiii] http://www.openauthentication.org/specifications

[lxiv] http://www.opennebula.org/

[lxv] http://www.cloudstack.org/

[lxvi] www.xen.org

[lxvii] http://www.microsoft.com/en-us/server-cloud/hyper-v-server/default.aspx

[lxviii] http://www.vmware.com/products/vsphere/esxi-and-esx/index.html

[lxix] http://www.linux-kvm.org/page/Main_Page

[lxx] http://fuse.sourceforge.net/

[lxxi] http://www.mysql.com/products/cluster/

[lxxii] http://cassandra.apache.org/

[lxxiii] http://ceph.com

[lxxiv] www.gluster.org

[lxxv] http://log2timeline.net/

[lxxvi] TPM is a specification defined by the Trusted Computing Group for chip that provides cryptographic operations such as key generation and storage, hashing and signing

[lxxvii] Secure coprocessors are tamper proof programmable hardware devices that may be attached to a computer to perform security related functions

[lxxviii] http://code.google.com/p/vmitools/