A Methodology for Designing Cloud Forensic-enabled Services (CFeS)

Abstract Cloud computing is used by consumers to access cloud services. Malicious actors exploit vulnerabilities of cloud services to attack consumers. The link between these two assumptions is the cloud service. Although cloud forensics assists in the direction of investigating and solving cloud-based cyber-crimes, in many cases the design and implementation of cloud services falls back. Software designers and engineers should focus their attention on the design and implementation of cloud services that can be investigated in a forensic sound manner. This paper presents a methodology that aims on assisting designers to design cloud forensic-enabled services. The methodology supports the design of cloud services by implementing a number of steps to make the services cloud forensic-enabled. It consists of a set of cloud forensic constraints, a modelling language expressed through a conceptual model and a process based on the concepts identified and presented in the model. The main advantage of the proposed methodology is the correlation of cloud services' characteristics with the cloud investigation while providing software engineers the ability to design and implement cloud forensic-enabled services via the use of a set of predefined forensic related tasks.

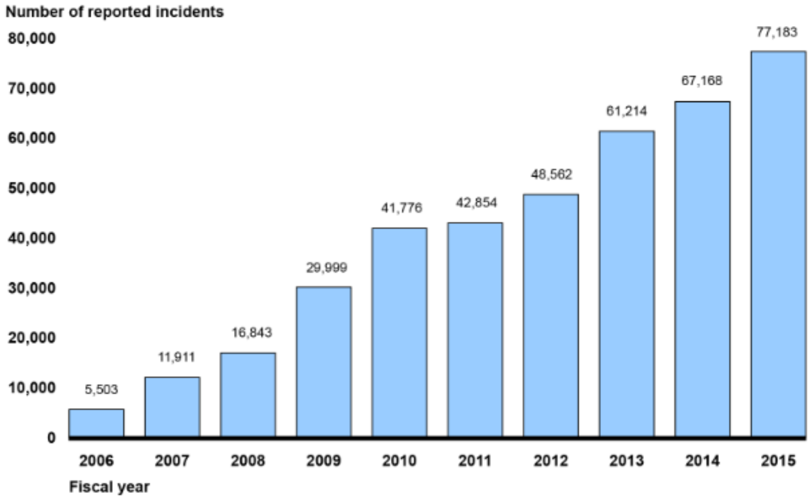
Keywords Cloud Forensics, Cloud Forensic Methodology, Cloud Forensic Process, Cloud Forensic Conceptual Model, Cloud Forensic Constraints.

1 Introduction

Over the past years, the technology of cloud computing has dominated the field of Information Technology (IT) by providing cloud services to consumers. Cloud's flexibility, the increasing demand for new cloud services and their high adoption by users are some of the reasons of this domination. By the end of 2016, an average organization uses 1,427 cloud services, an increase of 23.7% over the same period of the previous year [1]. However, cloud computing technology is one more field for criminal exploitation [2]. Software engineers responsible for the design and implementation of cloud services, in many cases, appear to forget or fail to pay the proper consideration on cloud forensic needs. This has a huge impact on a cloud forensic investigation due to the fact that the investigation cannot be conducted in a forensically sound manner. In order to deal with this issue and ensure that investigation standards are met, software engineers must comply with forensic standards and develop reliable cloud forensic-enabled services.

Perpetrators use cloud computing to gain access to information by exploiting vulnerabilities or they use cloud resources to distribute illegal context. In either case, they are trying to hide their real identity and keep their anonymity behind this "complex" environment. The number of incidents related to cyber-crime is a major issue among Cloud Service Providers (CSPs), consumers and Law Enforcement Agents (LEA) as it has been growing rapidly over the past few years. According to the United States Government Accountability Office (GAO), the number of cyber incidents affecting federal agencies has increased about 1,300 percent the last 10 years [3] (see Figure 1). In order to protect consumers from perpetrators, information system designers and software engineers should be able to design cloud forensic-enabled services that could assist investigators to solve cloud-based cyber-crimes. This is a great challenge for software engineers but also an opportunity to provide investigators with all the necessary capabilities to investigate an incident in a forensically sound manner. To accomplish the task, designers need to explore those forensic requirements and processes that will identify a cloud service as forensicable (in this paper the term forensicable is used to describe a service of being forensic-enabled).

After a thorough analysis of the respective literature, we came to the conclusion that there is a literature gap in supporting software engineers so as to identify forensic-related requirements for information systems [4]. Thus, to fill the aforementioned gap we present a requirements' engineering methodology to support the elicitation of forensic requirements. The methodology consists of a set of cloud forensic constraints, a modelling language expressed through a conceptual model and a process based on the concepts identified and presented in the conceptual model. The conceptual model presented in this paper not only includes the concepts that make a system forensic-enabled, but also the concepts for cloud forensic investigation identified in [4], raising the importance of the relation between a forensic-enabled system and an investigation process and how the latter is assisted when an incident occurs. In this way, an integrated conceptual model is produced to assist designers in a way that they will be able to design forensicable cloud services.



Source: GAO analysis of United States Computer Emergency Readiness Team and Office of Management and Budget data for fiscal years 2006-2015. | GAO-16-885T

Fig. 1 Incidents Reported by Federal Agencies, Fiscal Years 2006 through 2015.

The main contribution of this research is a) to identify and propose a set of forensic constraints introduced and expressed as feature diagrams that should be considered when designing cloud forensic enabled services, b) to propose a novel conceptual model that embodies all the necessary concepts required to design a forensic-enabled cloud system/service and which at the same time contributes to respective investigation procedures, and c) to present a process that engineers may follow for designing cloud service in forensic-enabled manner. These three interrelated parts compose an integrated conceptual methodology for designing cloud forensic-enabled services.

The rest of the paper is organized as follows. Section 2 presents the related work introduced by researchers concerning methodologies and frameworks in relation to digital and cloud forensic investigation, while section 3 introduces the identified forensic constraints described in a structured way. In section 4 a set of feature diagrams are described for expressing the basic tasks that need to be realized in a cloud service for becoming forensic-enabled. Section 5 presents the proposed conceptual model and the role of every concept in the design of cloud services. Section 6 presents a requirements engineering process that software engineers should follow when designing cloud forensic-enabled services. In section 7 the applicability of the methodology is examined in a real case study while section 8 discusses important issues raised from the specific research. Finally, section 9 concludes the paper by addressing useful remarks and issues for further study.

2 Related Work

Over the past years, a number of researchers introduced various methodologies, frameworks and models regarding the way of conducting proper forensic investigation both in digital and cloud environments. McKemmish [5] was one of the first researchers to define the term forensic computing (actual introducing the term digital forensics) and the definition given was “the process of identifying, preserving, analyzing and presenting digital evidence in a manner that is legally acceptable”. In 2001, the First Digital Forensic Research Workshop (DFRWS) [6] defined a generic investigative process that could be applied to the majority of investigations involving digital systems and networks. The model establishes a linear process and many researchers have used this framework to develop their own work. The same year, the U.S. Department of Justice introduced the Electronic Crime Scene Investigation: A Guide for First Responders [7]. It was developed to assist State and local law enforcement and other first responders who might have been responsible for preserving an electronic crime scene and for recognizing, collecting, and safeguarding digital evidence.

The Abstract Digital Forensic model [8] was based on DFRWS model and includes three more stages. It allows a standardized process to be defined without specifying the exact technology involved. The Integrated Digital Investigation Process (IDIP) [9] model introduced in 2003 is based on the crime scene theory for physical investigations. The model lends many of the same phases of the previous models, but it uses the theory that a computer is itself a crime scene. The Enhanced Digital Investigation Process model [10] is based on the IDIP model and separates the investigations in primary and secondary crime scenes, while depicting the phases as iterative instead of linear.

The Extended Model of Cybercrime Investigations introduced by Ciardhuain [11] in 2004, identifies the activities of the investigative process and the major information flows in that process, an important aspect of developing supporting tools. The model includes information flow description between different phases and it is considered as the most complete framework [12]. The Hierarchical Objectives Based Framework [13] for the digital investigations process in 2005, proposes a multi-layer, hierarchical framework, as opposed to the single-tier approach being presented to date. It includes objectives-based phases and sub-phases that are applicable to various layers of abstraction, and to which additional layers of detail can be easily added as needed. In 2006, the Forensic Process [14] proposed by National Institute of Standards and Technology (NIST), transforms media into evidence for law enforcement or for organization’s internal usage. The same year, Von Solms [15] introduced a control framework for digital forensics to provide a sound theoretical basis for digital forensics, as well as a reference framework for digital forensics governance within organizations.

The Digital Forensic Investigation Framework (DFIF) [12] simplifies the existing complex framework and it can be used as a general DFIF for investigating all incident cases without tampering the evidence and protects the chain of custody. In

2010, Digital Forensic Evidence Processes [16] is introduced defining that the steps should be executed in a manner that meet the legal standards of the jurisdiction and the case. The Systematic Digital Forensic Investigation Model [17] proposed in 2011, helps forensic practitioners and organizations to set up suitable policies and procedures. The proposed model places emphasis on the cyber-crime and cyber-fraud. The Harmonized Digital Forensic Investigation Process model [18] introduced in 2012, proposed several actions to be performed constantly and in parallel with the phases of the model, in order to achieve efficiency of investigation and ensure the admissibility of digital evidence. It is an iterative and multi-tiered model, where each phase contains a set of sub-phases.

In 2012, two similar models introduced based on the Forensic Process; the Forensic Investigations Process [19] in cloud environments and the Cloud Forensics Process [20]. Due to the evolution of cloud computing the stages were changed to apply basic forensic principles and processes. They both focused on the competence and admissibility of the evidence while keeping into consideration the human factor. The Integrated Conceptual Digital Forensic Framework for Cloud Computing [21] proposed in 2012, is based on [5] and [14] and it emphasizes on the differences in the preservation of forensic data and the collection of cloud computing data for forensic purposes. The Cloud Forensic Maturity Model (CFMM) presented in 2012 [22], is a reference model for evaluating and improving cloud forensic maturity. The model is a step forward towards an acceptable solution for cloud forensic investigation.

In 2013, Adams [23] introduced the Advanced Data Acquisition Model (ADAM) that can assist digital forensic practitioners when it comes to presenting evidence in court that originated in the cloud. It is a promising model taking into consideration lots of factors concerning digital and cloud forensic investigation. The Integrated Digital Forensic Process Model (IDFPM) [24] presented in 2013, is at the same time a merging of existing forensic models, an integration of them and a purification of the terminology used, resulting in an all-encompassing standardized IDFPM. In 2015, Zawoad et al. [25] proposed a cloud forensic process called Open Cloud Forensics (OCF) model. The proposed model can support reliable forensics in a realistic scenario by considering the important role of CSPs.

A detailed review at the respective methods has been also conducted and presented in [26, 27]. This analysis revealed the challenges and the open issues related to cloud forensics and their possible solutions. It has also highlighted that there is an urgent need for designing and developing new methodologies and frameworks to support cloud forensics. The review concludes that no methodology or framework has been developed to cover every aspect and every stage in a cloud forensic investigation. The new methodology is developed beyond the boundaries of the Requirements Engineering, providing a holistic approach of all the involved stages, from the need of “cloud services decision” through to the “evaluation”.

3 Forensic Constraints

This section presents a list of concepts that should be realized in order for a cloud-service to be characterized as cloud forensic-enabled. Thus, a list of concepts is presented following our previous review on the respective field [27]. The concepts presented are defined as constraints since their implementation forces the mandatory use of specific technologies in addition to the existing functionality of the services.

Forensic constraints are requirements related to system forensicability (in this paper we use the term forensicability as a system or a service that can be forensic-enabled; can be developed in a forensic sound manner) and specify a system's or service's quality attributes. To identify a set of cloud forensic constraints first we need to clarify the concept of cloud service. In fact, a cloud service is any resource made available to consumers over the Internet such as data storage, e-mail, web hosting, etc. CSPs are responsible for providing those services through service models and deployment models. Depending on the design and implementation, a cloud service may contain vulnerabilities that can be exploited by malicious actors [28]. These vulnerabilities are sometimes hard to avoid and may harm consumers. To investigate the incident in a forensically sound manner and find a solution, the implementation of the specific service should take into consideration various parameters related to forensic requirements. Authors in [29] introduce a forensics-friendly cloud computing architecture and state that "*we need to preserve logs, proof of data possession, provenance information and timestamp securely*" in order to support trustworthy forensics in cloud. On the other hand, evidence should be handed to users, protective actors, or court authorities whenever they asked.

For a service to be characterized as cloud forensic-enabled (meeting specific criteria) depends both on the people using the particular service and on the way it has been implemented. From the people's perspective, National Institute of Standards and Technology (NIST) highlights that the actors involved in the cloud are: consumers, providers, auditors, brokers and carriers [30]. Actors interact with one another depending on their roles in the cloud. The technical perspective focuses on the procedures, forensic mechanisms, security and private policies that are used to implement a cloud service in order to make it reliable and trustworthy to the people.

Based on the cloud characteristics and the forensic properties seven cloud forensic constraints have been identified from the respective literature [22, 29, 31-35]. These constraints have a lot in common with security and privacy concepts identified in various research works [28, 36-38]. Some of the concepts are identical in both worlds, especially when they are examined under the technical point of view. This is due to the fact that the cloud forensic process relies on the privacy and security capabilities to help resolve forensic issues. Based on our previous work [26, 27, 39] regarding the identification of forensic investigation stages, challenges and solutions, we present in the following tables various helpful information related to the identified cloud forensic constraints. The identification and analysis of the information presented in tables 1, 2 and 3 was a mandatory step for the proposed tasks presented in section 4.

Table 1 Forensic Constraints' Definitions and their applicability in the Forensic Investigation

Forensic Constraint	Definition	Stages
Accountability	The CSP's obligation to protect and use consumer's data with responsibility for its actions and liability in case of an issue	Identification, Preservation-Collection, Examination-Analysis, Presentation
Transparency	The condition where an entity can have full access and freedom to manage and control its own data in the cloud at any given time and allow feedback from the entities that accommodate it	Identification, Preservation-Collection, Presentation
Internal Disciplinary Procedures	The process through which a cloud provider or broker deals with its employees in order to ensure that its employees follow certain norms of discipline	Identification, Preservation-Collection, Examination-Analysis
Access Rights (Policies)	The permissions that are assigned by an administrator to grant users and applications access to specific operations.	Preservation-Collection, Examination-Analysis
Isolation	The mechanism to ensure that consumers' data is sealed and cannot be seen by other tenants	Preservation-Collection
Legal Matters	The procedures and actions that need to be under-taken related to jurisdiction issues, international law, contractual terms and constitutional issues	Identification, Preservation-Collection
Traceability	The ability, for the data to be traced or not by the user [28] and the capability of keeping track of the actions taken at any given point. It is also the ability to trace the activities of a consumer	Identification, Preservation-Collection, Examination-Analysis

Trust in the cloud is a very important notion and it could be identified as another forensic constraint besides the seven previously described. Trust is the customer's level of confidence in using the cloud. Due to the fact that trust is fulfilled through the identified forensic constraints it should be dealt in a holistic way and not be dealt independently. Implementing the forensic constraints and using them with cloud services automatically increases the customer's level of confidence. This in turn (making cloud forensic-enabled services) assists towards the implementation of trustworthy services.

The implementation of a service consists of numerous actions that need to be carefully examined to prevent malicious activities. These actions can be implemented using one or more forensic constraints. On the other hand, one forensic constraint can be used to implement more than one action in a cloud service. For example, when we take under consideration the storage cloud service, the authorization access, which is part of the access rights forensic constraint, can be used in different activities.

Table 2 Forensic Constraints related to Challenges and Cloud Actors

Forensic Constraint	Challenges	Actors
Accountability	Access to evidence in logs, Dependence on CSP-Trust, Service Level Agreement (SLA), Chain of custody, Documentation, Compliance issues	Consumer, Cloud Service Provider, Cloud Broker, Cloud Auditor
Transparency	Access to evidence in logs, Dependence on CSP, Physical inaccessibility, Service Level Agreement, Volatile data, Imaging, Documentation, Compliance issues	Consumer, Cloud Service Provider, Cloud Broker
Internal Disciplinary Procedures	Internal staffing-Chain of custody, Integrity and stability-Multitenancy and privacy, Service Level Agreement	Cloud Service Provider, Cloud Broker, Cloud Carrier
Access Rights (Policies)	Internal staffing-Chain of custody, Integrity and stability-Multitenancy and privacy, Time synchronization-Reconstruction, Identity	Consumer, Cloud Service Provider
Isolation	Integrity and stability-Multitenancy and privacy	Cloud Service Provider
Legal Matters	Access to evidence in logs, Service Level Agreements, Multi-jurisdiction-Distribution-Collaboration	Cloud Service Provider, Cloud Broker
Traceability	Client side identification, Volatile data, Identity, Time synchronization-Reconstruction	Consumer, Cloud Service Provider

Table 3 Forensic Constraints related to Solutions

Forensic Constraint	Solutions
Accountability	Ensure policies and standards are met with great responsibility and any problems arising from CSPs' actions are remedied promptly. Monitor data and logs with appropriate tools in order to satisfy the policies and demonstrate compliance [32]. Develop assurance methodologies. Obtain assurance of the services in cloud by using vulnerability assessment and penetration testing approaches [33].
Transparency	CSPs should provide consumers with the freedom to handle and control their own computation and data according to their usage. Strong SLAs should be built between the parties, and contract agreements should be signed. On the other hand, trusted mechanisms should be implemented to help establish a better relationship between parties and increase mutual trust.
Internal Disciplinary Procedures	Frequent personnel surveillance to prevent turning rogue and intentional or accidental compromise consumers' data. Well-trained and accredited personnel to undertake the sensitive parts of the investigation. Access rights both on physical equipment and digital data. Enforce legal contracts in employee behavior policy.
Access Rights (Policies)	Use security checkpoints. Enforce stringent registration and validation process. Make sure important updates are installed on time. Prohibit user credential sharing among users, applications, and services
Isolation	Separate data through partitioning. Ensure that memory, storage, and network access are isolated
Legal Matters	Global unity must be established. New regulations and international laws should be developed to secure forensic activities will not breach any laws or regulations under any jurisdiction. Accessing and handling data by third parties should be ensured and should be structured in a manner consistent with the provider's policies
Traceability	Enterprises should track deployment options from the data center to the business process to make sure the value chain is uncompromised. Track and store all the users' actions through logs. Data and client's traffic should be monitored at all times. Monitor Quality of Service (QoS) for SLAs regularly to determine any vulnerabilities. Users' activities and accounts should be monitored at all times.

A number of simple examples for better understanding the meaning of the seven forensic constraints is shown in Table 4.

Table 4 Indicative examples of the seven forensic constraints.

Constraints	Examples
Internal disciplinary procedures	Being able to prove CSP's personnel is trained, specialized and accredited to use sensitive data through contract agreements.
Transparency	Being able to trace deleted data in dropbox.
Accountability	CSP can provide information about deletion of data at any time.
Legal matters	Data being able to receive the same legal protections no matter it is stored on dropbox or on a personal computer.
Access rights	Only eligible users can have access to private information in dropbox.
Isolation	Being able to avoid contamination in case of an incident.
Traceability	Monitor all the actions taken from the access authorization to deletion of data.

4 Addressing Cloud Forensic Constraints

For each forensic constraint identified in the previous section, a feature diagram is introduced for expressing the basic tasks that need to be realized in order for every forensic constraint to be addressed. Feature modeling is very helpful since it assists engineers in modeling the properties of concepts and their interdependencies and organizing them into a coherent model referred to as a feature model [40]. A feature diagram consists of a set of nodes, a set of directed edges and a set of edge decorations. The edges along with the respective nodes form a reverse tree like goal models. The edge decorations are drawn as arcs connecting subsets or all of edges originating from the same node. Various types of connections exist depending on the connection of the sub features with the main feature. Mandatory, optional, alternative and or-feature types do exist. A survey on the meaning and available notations of feature models can be found in [41]. In this paper all feature diagrams use the mandatory sub-feature notation based as proposed by Czarnecki-Eisenecker in [40] since all sub-features should be realized in order for the main feature (in our case the respective forensic constraint) to be addressed. Figure 2 presents a template of the feature diagram used for representing in the form of sub-features the tasks that need to be accomplished per forensic constraint.

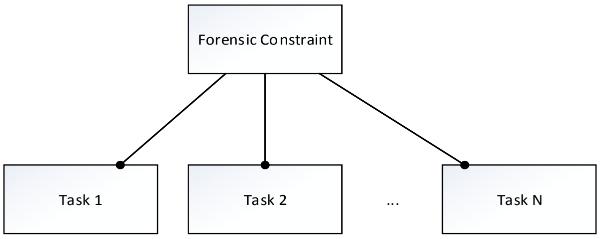


Fig. 2 A generic forensic constraint feature diagram

The proposed diagrams describe the actions a cloud provider should produce/take in order to make a cloud service forensic-enabled. The forensic constraints focus on the cloud provider side since it is the entity that owns the infrastructures and provides the cloud services to consumers. The tasks shown in each feature diagram refer to the cloud provider's activities, which they should have implemented, regardless the implementation order. Thus, the constraints and the tasks presented are executed on the provider's side. On the other hand, whenever a cloud service is implemented by a third party and a contract agreement is signed between the provider and the third party, it is the latter's obligation to comply with the forensic constraints in order to realize the cloud service as forensic-enabled. The same applies for the cloud brokers or any other entity involved. The cloud provider is entitled to reject any third party that refuses to comply with the fulfillment of the forensic constraints and can seek for another party who is willing to do so. For instance, if a provider offers a service to a consumer ensuring there is no problem with jurisdictions, the third party the provider relies on, should also ensure that no issues will arise. Hence, strong SLAs should be built and signed between the parties stating all the necessary details.

The feature diagram for the accountability constraint, shown in Figure 3, presents and describes the relevant tasks needed to be undertaken to ensure that the constraint is fulfilled. Cloud providers should ensure that strong SLAs will be signed between third parties/consumers and on the other hand, policies and standards are put in practice. Assurance is obtained by providing security certification or validation exercise such as ISO 27001 certification and the results of a SAS70 Type II audit [31]. All the actions undertaken by the provider, third parties and the consumers should be monitored so as to ensure that a prompt solution will be given in case of an incident. Attributability is provided in revealing which system element or actor is responsible in case of a deviation from the expected behavior [31]. In the case that one or more of the described tasks have not been fulfilled, the provider should seek or implement techniques that resolve the issues. The same applies for all the constraints listed in the paper.

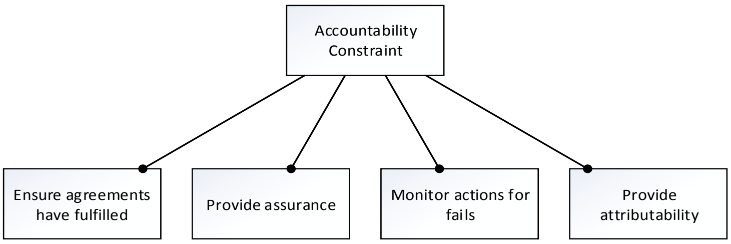

Accountability Constraint
Ensure agreements have fulfilled
Provide assurance
Monitor actions for fails
Provide attributability

Fig. 3 Accountability feature diagram.

The transparency feature diagram in Figure 4, highlights three tasks that should be implemented. CSPs need to ensure visibility of the applications by providing information about them at any time and inform consumers about the location/s of their data. They also need to notify the consumers about their procedures and policies on how the data is being treated. Finally, all CSPs need to be transparent. Notifications about the policy violations should be used to notify consumers in case of an incident.

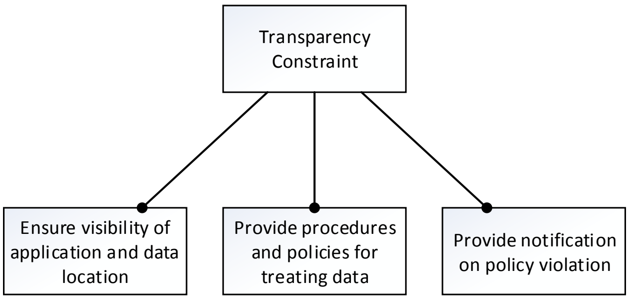

Transparency Constraint
Ensure visibility of application and data location
Provide procedures and policies for treating data
Provide notification on policy violation

Fig. 4 Transparency feature diagram.

The tasks a CSP needs to undertake to fulfill internal disciplinary procedures constraint are presented in Figure 5. Discipline rules need to be implemented and all the personnel should follow them. In case of any deviations, CSP should be able to discipline the responsible party without harming its interests. Access rights, both physical and digital should be categorized and their allowance should be granted accordingly. Contracts between the CSP and its personnel should be signed, stating all the details about misuse of information and the penalties.

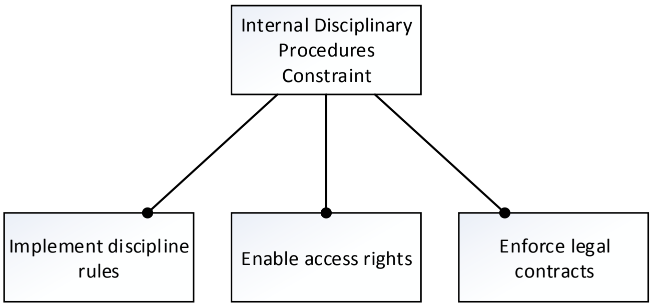

Internal Disciplinary Procedures Constraint
Implement discipline rules
Enable access rights
Enforce legal contracts

Fig. 5 Internal disciplinary procedures feature diagram

The access rights feature diagram in Figure 6 shows the tasks a CSP needs to implement to use the constraint. First, registration should provide all the necessary user's details and a control mechanism should validate the registration form to link as much information as possible with the user's true ID. Authentication and authorization control should be used to verify and determine the level of access of the users. Finally, access control should be implemented to enforce resources' required security.

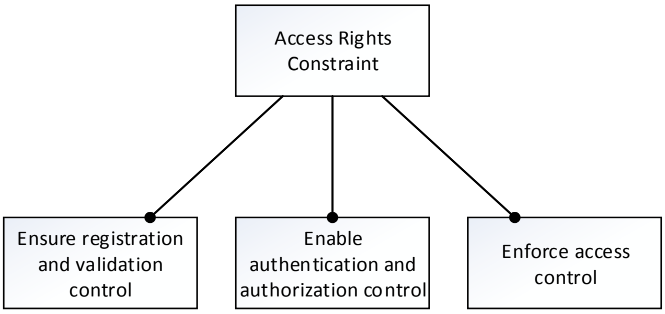

Access Rights Constraint
Ensure registration and validation control
Enable authentication and authorization control
Enforce access control

Fig. 6 Access rights feature diagram.

The isolation feature diagram in Figure 7 aims to ensure that a user does not have the right to access other users' data and that the data is securely stored. User's virtual machines are separated from the rest of the VMs and in case of an incident, contamination of other users is prevented. Privacy and confidentiality should be maintained at all times in such multi-tenant environment.

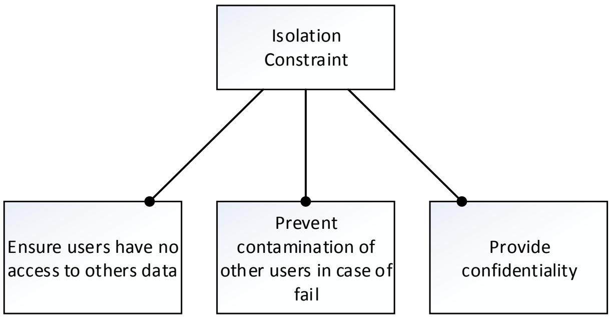

Isolation Constraint
Ensure users have no access to others data
Prevent contamination of other users in case of fail
Provide confidentiality

Fig. 7 Isolation feature diagram.

Legal matters feature diagram in Figure 8 is of vital importance since it is the most difficult to implement with all the different people, countries and laws involved. First, a strong and detailed SLA should be presented to ensure the terms of using cloud infrastructures. Then, ensure that a consumer's data should remain within the geographical boundaries of the country the user belongs to. Also, ensure that consumer's data should remain under the same jurisdiction and will not be distributed around the world. Finally, CSPs should hire and maintain specialized personnel on domestic/international laws and legislations related to cloud computing and data handling. The personnel should be trained on a regular basis to be brought up-to-date with new technologies.

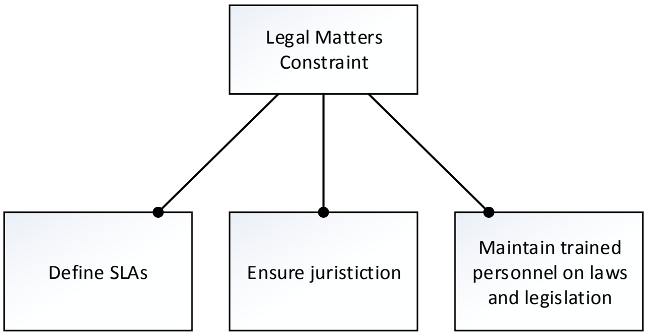


Fig. 8 Legal matters feature diagram

Traceability feature diagram in Figure 9 concerns users and their data. Monitoring users’ actions is important in order to reveal any faults. On the other hand, monitoring data logs and taking regular backups can reduce time and effort that is required to resolve malicious incidents. All logs should be stored and secured in places with limited access. The CSP should implement procedures to link data logs with a specific user and his/her activities.

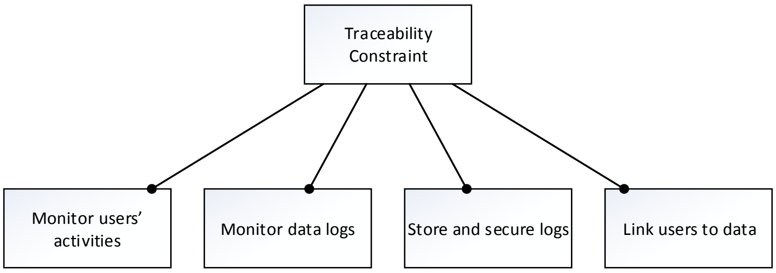


Fig. 9 Traceability feature diagram.

Each cloud forensic feature diagram introduces a set of tasks that need to be satisfied in order for a forensic constraint to be addressed. All the tasks should be applied to implement the forensic constraint, in any other case the forensic constraint cannot meet the forensic standards. Thus, each feature diagram has only one valid configuration and no alternatives can be introduced when applying it on every cloud service. These seven feature diagrams will be used later, in the methodology process, to match the activities of the cloud services’ with the tasks that need to be implemented as introduced by the feature diagrams. In this case, if a cloud service activity diagram includes the tasks introduced in an all feature diagrams then the service could be defined as forensic-enabled. As mentioned in the previous section, all the aforementioned seven cloud forensic constraints should be applied on a cloud service in order to be forensic-enabled.

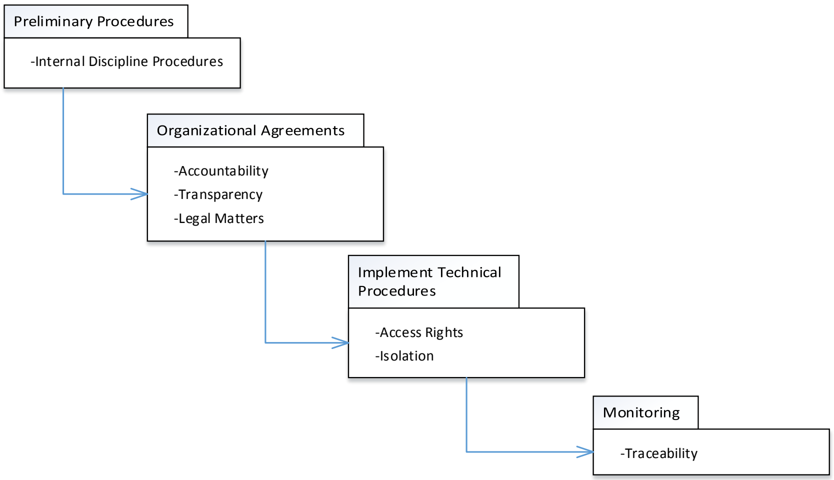


Fig. 10 Forensic constraints categorization and sequence.

Figure 10 presents the categorization of the seven constraints in four groups along with the correct order that the CSP should follow during the implementation of each constraint. The first category is the *preliminary procedures* and includes the internal disciplinary procedures constraint. This constraint should be implemented before all others, since companies need to establish and implement strong disciplinary procedures for internal usage. The second category is the *organizational agreements*, where the three forensic constraints of accountability, transparency and legal matters are included. This category deals with the agreements need to be signed and clarified between the provider and the clients or third parties. The next category is the *implement technical procedures*. This one includes two forensic constraints, the access rights and the isolation. The specific forensic constraints need to be implemented after the contracts are signed between the parties in order to know by which terms they will be implemented. Finally, the fourth category is the *monitoring*, in which the traceability forensic constraint is included. This is the last constraint in sequence that needs to be established since monitoring occurs after the implementation of the whole system or service. The proposed sequence is mandatory to follow and provides an important guidance to software engineers regarding the successful realization of the proposed constraints in the organizational processes.

The tasks used for each one of the proposed feature diagrams provide a generic approach for the fulfillment of the constraints. In this case, the constraints can also be applied in various cloud environments providing the proper level of technicality without being narrowed in one specific field.

5 Modelling language for Designing Cloud Forensic-enabled Services

The use of cloud computing for storing sensitive data raises concerns about the forensic investigation process in case of an incident. Forensic investigation in cloud computing requires a different approach from the traditional forensic process. This approach should take under consideration not only the technical, organizational and legal aspects but also the software engineer's requirements and the investigators' perspective. In order to produce a requirements engineering methodology to support the elicitation and modeling of the aforementioned forensic constraints, a common modelling language is introduced. The modelling language is presented in terms of a conceptual model, based on the concepts and the forensic constraints identified for designing a cloud forensic-enabled system. The conceptual model presented in this paper not only includes the concepts that make a system forensic-enabled but also the concepts for a cloud forensic investigation process from our previous work [4]. In this way, an integrated model is produced to assist designers in creating cloud forensic-enabled services considering the respective investigation requirements in the case of an incident.

Taking under consideration the forensic constraints identified, we proceed in identifying the concepts from the software engineer's perspective in order to develop a cloud forensic-enabled conceptual model. The model illustrated in Figure 11 shows a high-level map of the domain of Cloud Forensics. The model is based both on the concepts that make a system forensic-enabled and on the concepts that form a cloud forensic investigation process. In the model, the two different groups of concepts are clearly defined and separated from each other since they are used differently in the cloud forensics. On the other hand, some concepts that form the two groups are related to each other, thus the relationships between them must be clarified.

The first group (located in the main area of the conceptual model) shows the concepts related to a cloud forensic-enabled service. The second group (located on the upper right corner of the conceptual model, framed with dots) shows the concepts related to the investigation of an incident. The two groups have a common goal; the design of cloud forensic-enabled services in order for the investigators to solve an incident in a forensically sound manner. Once the process of making a system forensic-enabled is implemented and the cloud forensic investigation process is developed, then, protective actors just need to follow the respective steps.

In the next paragraphs, a detailed presentation of the two groups of concepts is introduced, describing all the aspects that will assist software engineers in designing a cloud forensic-enabled system/service and investigators to solve an incident in a forensically sound manner.

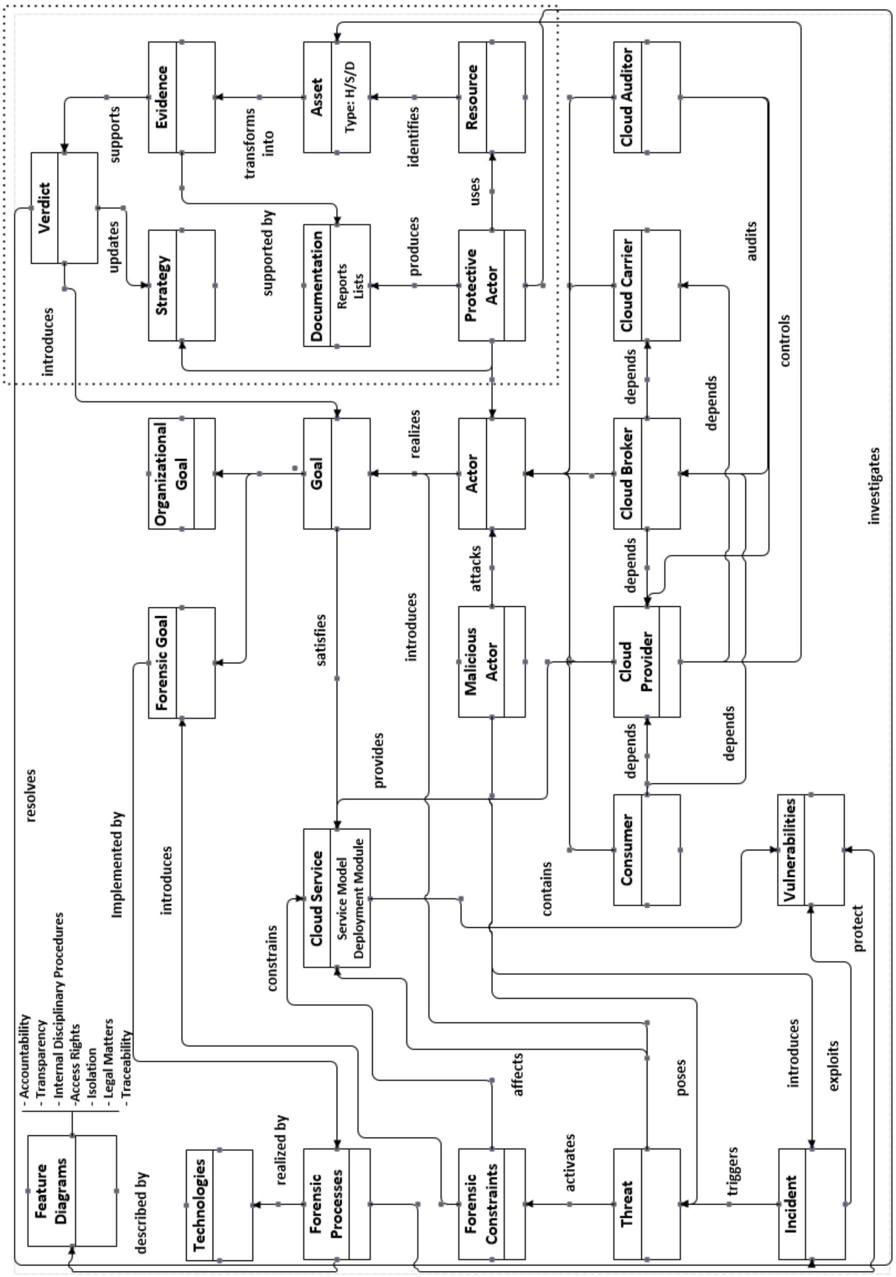


Fig. 11 Cloud Forensics Conceptual Model

5.1 Concepts related to cloud forensic-enabled system

As mentioned earlier in the paper, there are two different groups of concepts concerning the cloud forensic process. The first group assists software engineers in designing and implementing trustworthy cloud services. It describes all those concepts a designer needs to include in his/her design to produce a forensic-enabled service. The list of the concepts is as follows:

Actor: According to NIST [30] the actors involved in the cloud are: consumers, providers, auditors, brokers and carriers. The definitions given for the five actors are as follows:

Cloud Consumer: "*Person or organization that maintains a business relationship with, and uses service from Cloud Providers*" [30]. A consumer can be any person that uses the cloud either as a common user or as a malicious user. The malicious actor is the one who introduces an incident and he/she is responsible for attacking any other actor involved in the cloud. He/she uses CSPs' services to launch his/her attacks exploiting vulnerabilities hidden behind anonymity. Consumers have dependencies on both cloud providers and cloud brokers.

Cloud Service Provider: "*Person, organization or entity responsible for making a service available to interested parties*" [30]. CSPs are responsible for offering multiple services to consumers through their deployment modules and service models. Their major concern is to rent as many services to clients as possible. Their services should be supplied with responsibility and reliability according to service level agreements signed between actors. CSPs depend mostly on cloud carriers.

Cloud Broker: "*An entity that manages the use, performance and delivery of cloud services and negotiates relationships between Cloud Providers and Cloud Consumers*" [30]. The broker helps the consumer find the suitable cloud providers and negotiate contracts with them. The brokers' main dependencies are on CSPs and cloud carriers.

Cloud Carrier: "*An intermediary that provides connectivity and transport of cloud services between Cloud Providers and Cloud Consumers*" [30]. Cloud carriers are mostly traditional telecommunication providers responsible for delivering cloud services over their own network and other access devices. The carrier's main objective is to provide CSPs with secure and dedicated connections through service level agreements. In some cases, a cloud carrier can play the role of cloud provider at the same time.

Cloud Auditor: "*A party that can conduct independent assessment of cloud services, information system operations, performance and security of the cloud implementation*" [30]. Auditors are responsible for evaluating cloud providers' and brokers' services by performing audits in order to verify if their performance and security mechanisms are acceptable to the consumers.

Goal: The concept of the goal introduced in this model focuses on the realization and achievement of specific objectives, such as the way the system is designed, implemented, and operated. A goal can be either organizational or forensic. "*Organizational goals express the main organization objectives that need to be satisfied* *by the system into consideration*" [42]. Forensic goals are generated by forensic constraints. In cloud computing, when system engineers develop a service, they need to realize different forensic goals in order to make the service forensic-enabled. These forensic goals are being introduced by specific forensic constraints and are implemented within the use of forensic processes (explained in the next paragraphs). A goal or a number of them can satisfy a cloud service.

Cloud service: A cloud service is any resource made available to users over the Internet. Cloud Service Providers are responsible to provide those services through service models (IaaS, PaaS, and SaaS) and deployment models (public, private, hybrid, and community). Attackers exploit vulnerable services, thus is the most important asset along with the respective resources providing this asset.

Vulnerabilities: A vulnerability is a weakness in design, implementation or operation of a system/service that allows malicious actors to exploit the system/service, and create an incident in order to take control, breach, or violate the system/service. Cloud services may have one or more vulnerabilities that may compromise the integrity or privacy and security of the service. In order to be able to design forensic-enabled services and mitigate the respective vulnerabilities appropriate forensic processes need to be implemented.

Incident: "*A breach of security or a loss of integrity that has impact on the operation of network and information system core services, which public administrations and market operators provide*" [43]. The malicious actor is responsible for introducing an incident in order to exploit vulnerabilities of cloud services. On the other hand, the incident triggers threats for the system. Protective mechanisms should be implemented based on previous incidents to assist software engineers to develop forensic-enabled services.

Threat: A threat is an action that might cause harm to a system/service. Malicious actors pose threats to a system/service and these threats are triggered by their incident. Depending on the type of threat, specific forensic constraints are activated to deal with them. The threat aims to affect cloud services in order to gain control of specific assets.

Forensic constraints: Forensic constraints are non-functional requirements that relate to a system's/service's ability to be forensic-enabled and specify the system's or service's quality attributes. Forensic constraints identified and presented in the previous section allow software engineers to develop forensic-enabled systems/services; systems/services whose architecture supports forensic investigation. The forensic constraints should be applied to cloud services in accordance to the criticality of the service so as to guarantee the forensics. These constraints are being activated by the threats triggered by an incident and their main objective is to introduce and produce forensic goals.

Forensic processes: A forensic process is a mechanism, which handles evidence in a forensically sound manner, on one hand and on the other, determines what the vulnerabilities of the system/service are, so as to protect the system/service and meet the forensic goals introduced by forensic constraints. After identifying the potential vulnerabilities of the system/service, the most appropriate forensic process to attend to the specific vulnerability will be selected to eliminate the threat and make the service forensic-enabled. Forensic processes are realized with the help of technologies and described by feature diagrams.

Technologies: Technologies are these techniques and solutions used to handle digital evidence (identify, collect, preserve, analyze and present) and achieve protection in cloud systems. Techniques such as registration and validation that allow us to have accurate information about users, or logging and monitoring mechanisms that provide us information at all-time about users' activities. These procedures will be automatically performed to eliminate potential threats.

5.2 Concepts related to cloud investigation

The second group of concepts provides Law Enforcement Agents with the ability to understand all those concepts that are involved in a cloud forensic investigation and the importance of their roles. This is of vital importance since the cloud forensic-enabled service should be designed in a manner that the identified information will assist the investigator when an incident occurs. Thus, the concepts described in the proposed model should be able to collaborate with the information required during an investigation. The list of the concepts related to the investigation process that is considered in the proposed conceptual model is as follows:

Protective actors: Protective actors are the people (team) responsible for investigating an incident and trying to solve it. They conduct the investigation "*by utilizing and managing the forensic capabilities within the cloud environment adding their own forensic capabilities*" [22]. Protective actors use resources and develop strategies concerning decisions they have to take, based on the training, planning and preparation activities. Planning and organizing an actor's next moves in case of an incident, is very productive when the time comes. A well-organized preparation can improve the quality and availability of digital evidence collected and preserved, while minimizing cost and workload [44].

Resources: Protective actors use resources (personnel, tools, trainings plans, methods, etc.) to resolve an incident. The resources that can be used related to personnel are the technicians (provider, protective actor or victim), the law officers and everyone else working on the case. Using the resources in a proper way the investigation can move forward since the resources can identify all the assets (especially data) hidden in the cloud environment.

Assets: CSP is the one who controls all the assets during a forensic investigation. There are three types of assets; hardware, software and data. Investigators extract data from media and identify traces in data. According to [14] the forensic process transforms media into evidence in three steps. First data is extracted from media and transforms it into a new format, then data is transformed into information and finally, information is transformed into evidence. The types of assets that can be transformed to evidence include, but are not limited to, remote computers, hard discs, deleted files, times and dates associated with modifications, computer names and IP addresses, usernames and passwords, web server logs, windows event logs, registry entries and temporary files, browser history, temporary internet files and cache memory, etc. Assets related to cellular phones could be SIM cards, call logs, contacts, SMS and MMS, calendar, GPS locations and routes.

Evidence: Evidence is the most important concept of the legal system. Depending on the way evidence has been acquired and handled in order to maintain chain of custody it can be admissible or not, in a legal proceeding. The collection of the assets with the use of appropriate resources may lead to the identification of useful evidence. Examining and analyzing the assets with the use of software tools can help investigators to find evidence and build a case in a court of law. Documentation supports the evidence and the strongest type of evidence obtained can support an assertion and pursue a positive verdict.

Documentation: The main objective of documentation is to keep the investigation properly documented so as to increase the chances of winning a case in a court of law or in an internal investigation. Documentation at the early stages of the incident also helps to keep track of all the actions having been taken and to proceed with different techniques. Any risk analysis or assessment tests performed during the training and preparation should be documented to assist the team. All tools, processes, methods and principles performed should be documented properly in order to maintain the chain of custody. Any changes made to the evidence should also be recorded. According to [45], "*a properly maintained chain of custody provides the documentary history for the entire lifetime of evidence discovered during an investigation*". To present the evidence in a court as admissible, all the parties (staff, CSPs, third parties) that have conducted the investigation should record their actions through logs and notes e.g., who handled the evidence, how it was done, was the integrity of the evidence maintained, how it was stored, etc.

Strategy: Strategy is developed both by protective actors and by consumers. As far as protective actors are concerned, this concept deals with the methods and policies they use to proceed in an investigation. Protective actors have to take decisions about the acquisition of evidence or the presentation. The outcome of the trial depends on their decisions. On the other hand (consumers point of view), strategy plays a vital role in the preparation and planning of the system to meet the organizational goals. Training is also part of an organization's strategy in order to support forensic services and be prepared to handle an incident.

Verdict: This concept is related to the evidence and particularly to its presentation. When the verdict is announced, the incident is either resolved or an appeal follows. Either way, the strategy should be revised and updated to identify areas of improvement and review methodologies and procedures. Even though verdict as a concept does not belong to a cloud forensic investigation (a verdict is a judgment in a court of law, not a protective actors action) we strongly believe that it must be illustrated in the model. This is due to the fact that the decision of a jury concludes (closes) a forensic investigation. It is the outcome of the investigation whether it is positive or negative.

The two groups of concepts shown in the conceptual model interact with each other in order to produce a model that represents a holistic solution to the cloud forensic investigation problem. This could be achieved by implementing (the software engineers) cloud forensic-enabled services to assist investigators with cybercrimes. Table 5 presents an instantiation of all the concepts used in the conceptual model. This instantiation assigns a value to each one of the concepts. The scenario where the instantiation is based is the following:

An executive member (consumer) of an organization stores sensitive data in the cloud using Microsoft Azure as a CSP. A malicious actor who uses the same provider exploits vulnerability in the system and steals the data from the consumer. LEAs have been called to trace and find the malicious actor in a forensically sound manner.

Table 5 Instantiation of concepts.

Concepts	Instantiation of concepts
Malicious Actor	A user who wants to steal information
Consumer	Member of the Organization
Cloud Provider	Microsoft Azure
Cloud Broker	Netskope
Cloud Carrier	AT&T
Cloud Auditor	StarAudit
Goal	Provide storage capabilities to organization's members
Cloud Service	Data storage platform in cloud
Vulnerabilities	Failure to provide isolated storage service to consumers
Incident	Sensitive data have been stolen from consumer
Threat	Data Leakage
Forensic Constraint	Traceability
Forensic Processes	Store data in the cloud providing monitoring capabilities
Technologies	Data and operation logs tracing
Protective Actor	Law Enforcement Agents
Resources	Forensic tools, LEA's and CSP's personnel
Assets	Card payment information, CSP's subscriber id, logs, virtual machine and storage data, usernames and passwords
Evidence	IP address, username and password, logs
Documentation	Action plan report, methodology report, resource report, assets report, evidence report
Strategy	LEA acquires evidence through monitoring and snapshots
Verdict	Strong evidence brought a conviction

6 Methodology Process

The next step to the completion of the methodology is to develop a process based on the concepts identified and presented in the conceptual model. The process should be in accordance with the organization's needs. [46] examined the role of business process modelling (BPM) techniques in Information Systems security analysis and design (IS-SAD) and presented a generic framework for IS-SAD. They stated that the BPM technique should support tasks such as:

- *analyze the organization*
- *select the systems to be examined*
- *identify and analyze threats and vulnerabilities*
- *identify and evaluate entities that need protection*
- *design secure processes*
- *assess countermeasures' effectiveness and efficiency*
- *develop a security policy*

The area of business process modeling offers novel insights in the requirements engineering world since it offers solutions that combine the organizational-based activities along with modeling activities. The work presented in [47] is a more recent example of a business process approach that deals with security issues. In this paper, we prefer to follow this path by adopting IS-SAD. The scope is to bridge the gap between the stakeholders and the forensic analysis. Currently, compliance with forensic investigation is conducting through the selection and enforcement of security technologies selected in an ad-hoc way. By providing a structured methodology to stakeholders and software engineers, for the former to identify their actual needs prior to implementation and for the latter to be able to transform stakeholders' needs with a structured and robust way into feasible solution, is the goal of the proposed methodology.

The tasks listed in the previous paragraph have been considered and they can be used as a preliminary step in order to implement our process. The process itself provides the necessary steps towards a cloud forensic-enabled system/service based on the potential vulnerabilities of the system/service and the systematic analysis of forensic requirements. On one hand, it assists in the identification of the organizational strategy and needs and on the other, it analyzes in depth the various organizational cloud services in order to provide the necessary requirements for well-structured cloud forensic-enabled services. The process consists of three main stages: *Organizational Analysis*, *Cloud Forensic Requirements Analysis*, and *Evaluation-Assessment*. Figure 12 illustrates the proposed process with its stages, steps, inputs and outputs.

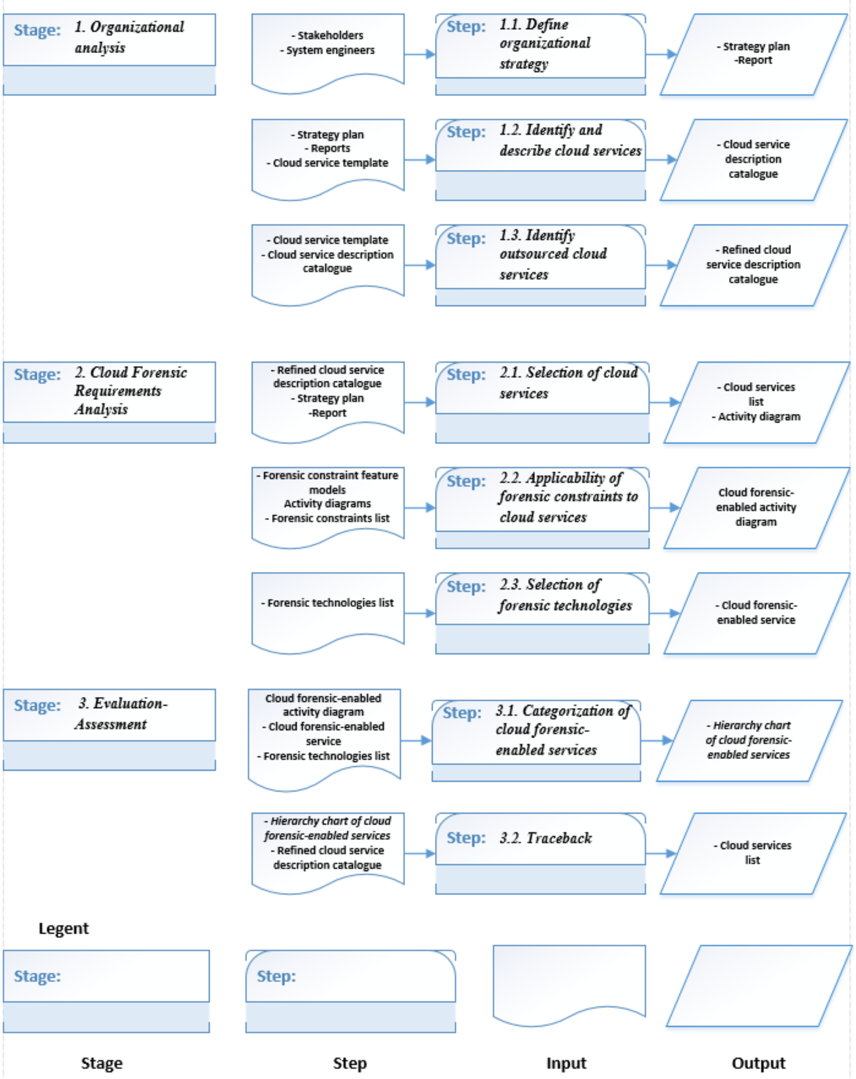

Stage: 1. Organizational analysis
- Stakeholders
- System engineers
Step: 1.1. Define organizational strategy
- Strategy plan
-Report
- Strategy plan
- Reports
- Cloud service template
Step: 1.2. Identify and describe cloud services
- Cloud service description catalogue
- Cloud service template
- Cloud service description catalogue
Step: 1.3. Identify outsourced cloud services
- Refined cloud service description catalogue
Stage: 2. Cloud Forensic Requirements Analysis
- Refined cloud service description catalogue
- Strategy plan
-Report
Step: 2.1. Selection of cloud services
- Cloud services list
- Activity diagram
- Forensic constraint feature models
Activity diagrams
- Forensic constraints list
Step: 2.2. Applicability of forensic constraints to cloud services
Cloud forensic-enabled activity diagram
- Forensic technologies list
Step: 2.3. Selection of forensic technologies
- Cloud forensic-enabled service
Stage: 3. Evaluation-Assessment
Cloud forensic-enabled activity diagram
- Cloud forensic-enabled service
- Forensic technologies list
Step: 3.1. Categorization of cloud forensic-enabled services
- Hierarchy chart of cloud forensic-enabled services
- Hierarchy chart of cloud forensic-enabled services
- Refined cloud service description catalogue
Step: 3.2. Traceback
- Cloud services list
Legent
Stage:
Step:
Stage
Step
Input
Output

Fig. 12 Forensic requirements engineering process for cloud forensic-enabled services

6.1. *Organizational analysis*

The first stage of the proposed process focuses on the presentation of the organization's goals and policies and in parallel produces an illustrated map (full description) of all cloud services the organization provides. This map assists the system analyst who is responsible for the migration of one service/system to the cloud, to identify and explore the needs, goals and structure of the organization in order to develop and implement the new system/service. This stage consists of three different steps.

6.1.1 Define organizational strategy

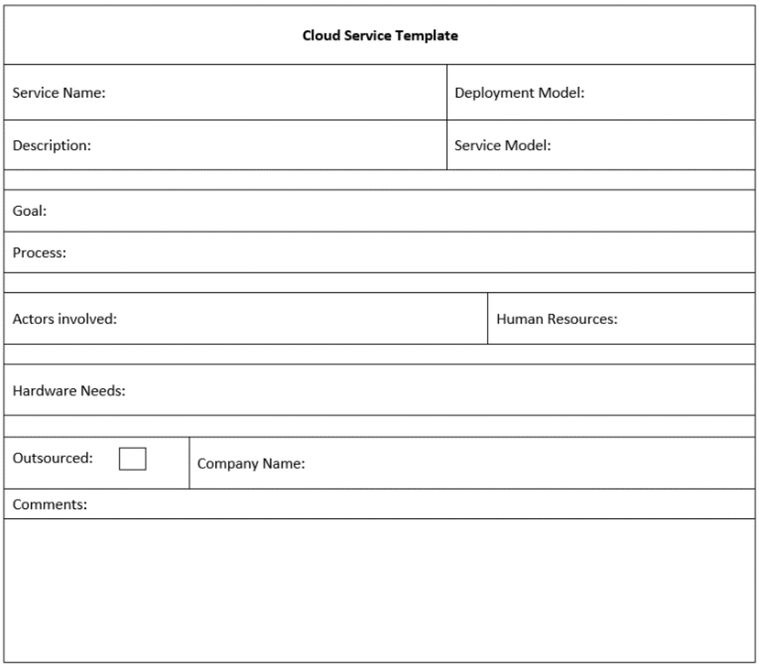
The first step of the process is to define organizational strategy, the actions a company intends to take in order to achieve its goals. The scope of this action is to be competent and reliable in the market. It is of vital importance to assess and evaluate not only the organizational goals in order to set the organizational needs, but also the consequences in the case those goals are not met. In order to design and implement a system, analysts should be fully aware of the structure of the organization itself. Organizational entities such as actors, goals, assets/infrastructure, resources, strategy and services should be identified and defined. Actors, responsible for the system and goal setting, should present their requirements and clarify all the aspects that will fulfill their needs. Stakeholders and software engineers need to implement a strategy plan about their cloud services to make them more competitive by producing cloud forensic-enabled services. They have to compare current circumstances with overall objectives to develop services that need improvement later in the process. On the other hand, the system analyst, responsible for the migration of the system, should be capable of understanding organizational strategy and needs to accomplish and develop a realistic plan. The output of this step is a report describing the organizational strategy on the aforementioned pillars.

6.1.2 Identify and describe cloud services

During the next step of the process, all cloud services provided to consumers should be presented and analyzed in order to understand the operation of the system. The presentation of cloud services should be thorough and a full analysis of each service should be provided separately. This analysis will contain the name of the service, a description, the deployment and the service model, which is applied to, goal objectives, storage needs, third parties, and all the aspects that an analyst needs to know about the nature of every cloud service. For the analysis of every service, a cloud service template will be used as input with all the necessary fields as shown in Figure 13. This action will assist the analyst to develop a global view of the infrastructures of the organization before the new set of requirements is designed and approved. The output of this step is the development of a description catalogue for each cloud service the organization provides.

6.1.3 Identify outsourced cloud services

The third step of this stage covers the outsourced cloud services that an organization might have. Third parties such as cloud providers, brokers, etc. provide a number of cloud services to organizations, to support their needs, both on a technological and infrastructural point of view. Some of them are specialized in a specific area making them more competitive in the market. The pattern followed is the same as in the previous step. Contracts and service level agreements, signed between the organization and third parties, will be reviewed and used, together with the description catalogue from the previous step to record all the necessary information for the outsourced cloud services. The output of this step is the development of a new refined description catalogue for each outsourced cloud service.



Cloud Service Template

Service Name:	Deployment Model:
Description:	Service Model:
Goal:	
Process:	
Actors involved:	Human Resources:
Hardware Needs:	
Outsourced: □	Company Name:
Comments:	

Fig. 13 Cloud Service Template

6.2. *Cloud forensic requirements analysis*

The next stage in the proposed process is the cloud forensic requirements analysis, which aims, first to identify the cloud services that an organization is willing to make forensic-enabled, and second to apply forensic constraints and technologies in order to do so. This step is concentrating on the organization’s services that need to be forensic-enabled, once they are given for public use in the cloud. It is an important stage and relies mostly on a well-structured design of the operation of each cloud service. Once the design of cloud services is developed and forensic constraints and technologies are applied, the organization has a full picture of the forensic requirements of each cloud service. This stage consists of three different steps.

6.2.1 Selection of cloud services

This step involves the selection of specific cloud services identified from the previous stage. The organization's stakeholders and software engineers, together with the analyst, will proceed to the selection of those cloud services that will be implemented in order to become forensic-enabled. This selection should be carried out in relation to the organizational strategy and goals defined earlier in the process. There will be a prioritization of cloud services depending on their importance to organization and a list with those services will be produced. This action can also involve the selection of the entire set of cloud services depending on the organization's budget, resources, etc. After the selection of cloud services, an activity diagram for each service will be generated, illustrating all the activities, actions and dependencies of the service. This diagram will assist the analyst to reason about the degree of forensicability of the service based on the forensic related activities existing in the implementation of the service.

6.2.2 Applicability of forensic constraints to cloud services

Within this step, it is important to capture the vulnerabilities and threats of each cloud service the organization wants to implement as forensic-enabled and apply the identified forensic constraints following the tasks described in each feature diagram before. The activity diagram of each selected cloud service, generated in the previous step, will be used as input and based on the forensic constraints feature diagrams the missing tasks will be identified. The application of the missing tasks in the form of activity will provide the necessary modifications leading to the transformation of the respective service as "forensic enabled". We have to take into account that the organization's software engineers may have already implemented some of the forensic constraints in order to make cloud service reliable and secure for public use. Nevertheless, if some forensic constraints are missing, the cloud service cannot be characterized as cloud forensic-enabled. The output of this step is a refined activity diagram with all forensic constraints identified and illustrated.

6.2.3 Selection of technologies

This step aims to identify and apply technologies that support the implementation of the forensic process. A number of technologies have been identified in the literature to support forensic requirements. The selection of the technology, which will be used, depends on a number of factors, such as the actors involved, the resources and the technical complexity. From the actors' perspective, it involves mainly the forensic engineers that will implement the technologies and the stakeholders. As far as the resources are regarded, they depend on the organization's financial capability. From a technical perspective, there are specific steps that need to be followed:

- The Cloud Service Template is used as input to identify two important aspects: the deployment model that the cloud service is applied to and its service model. These two characteristics can help forensic engineers to select only the technologies that concern the specific characteristics excluding the ones that are not applicable.
- The cloud forensic-enabled activity diagram for each cloud service is taken as input to observe the number of the forensic constraints that are not satisfied and they need to implement. The suggested technologies concern only the forensic constraints that are not satisfied.

Figure 14 illustrates the necessary steps that need to be taken in order to identify and select the technologies for the implementation of forensic constraints. When technologies are applied to activity diagrams, a new service is implemented which is cloud forensic-enabled and ready to support a cloud forensic investigation.

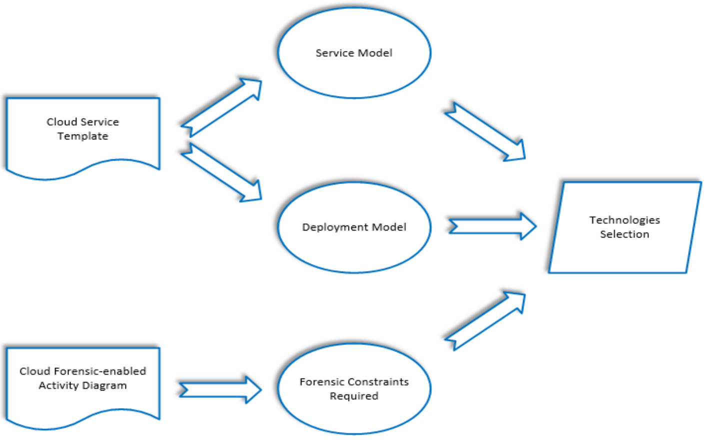


Fig. 14 Important steps for the selection of technologies

For suggesting the adequate technologies per forensic constraint, taking as input the deployment model, the service model and the missing forensic-related constraints, we have grouped a list of possible solutions on our previous papers that categorize the existing solutions based on these criteria [26]. A snapshot of this table is shown in Table 6. The specific categorization is very important and can assist us on automating the suggestion of respective technical solutions based on the aforementioned criteria.

Table 6 Snapshot of a list of possible solutions.

Cloud Forensic Challenges	Solution	Private	Public	IaaS	PaaS	SaaS
Access to evidence in logs	Secure-Logging-as-a-service (SecLaaS) mechanism	√	√	√	√	√
	Status data extraction and checking	√	√	-	√	-
	Log management architecture	-	√	-	-	√
	Logging mechanism	√	√	-	√	-
	Log-based model	√	√	-	√	√
	Digital forensic readiness model	√	√	√	√	√
	Management plane	-	√	√	-	-
	Logging framework	√	√	√	√	√
	Eucalyptus framework	√	√	√	-	-
Dependence on CSP - Trust	Accountable cloud	√	√	√	√	√
	TrustCloud framework	√	√	√	√	√
	Eucalyptus framework	√	√	√	-	-
	Trusted Third Party (TTP)	-	√	√	√	√
	Layers of trust model	-	√	√	-	-
Service Level Agreement (SLA)	Well and clear-written terms	√	√	√	√	√
	External auditors	√	√	√	√	√
	Service guarantee, violation detection, credit and standardization	-	√	√	√	√
	Trusted timestamping	√	√	√	√	√
	QoS and SLA model	-	√	√	√	√
Integrity & stability - Privacy & multi-tenancy	Digital signature	√	√	√	√	√
	Trusted Platform Module (TPM)	√	√	√	√	√
	Digital forensic readiness model	√	√	√	√	√
	Distributed signature detection framework	√	√	√	√	√
	Multi-tenancy model	√	√	-	-	√
	Proofs Of Retrievability (PORs)	√	√	√	√	√
	Data entanglement approach	√	√	√	√	√
	Entangled encoding scheme	-	√	√	√	√
	Trusted Cloud Computing Platform (TCCP)	√	√	√	-	-
	Secure role-based access control	√	√	√	√	√
	Identity and access management in future internet architecture (IAMFI)	√	√	√	√	√
	Data access control for multi-authority cloud storage (DAC-MACS)	√	√	√	√	√
	Provenance system	√	√	√	√	√

6.3. *Evaluation-Assessment*

The last stage of the process is the evaluation-assessment of cloud forensic-enabled services. During this stage, stakeholders decide which of the cloud services will be implemented according to their strategy and budget. A thorough study of the results produced in the previous stages is taking place and an assessment of the organization strategy is re-evaluated. The stage consists of two steps.

6.3.1 Categorization of cloud forensic-enabled services

After the development of the refined activity diagram and the selection of appropriate technologies per selected cloud service, a hierarchy chart of cloud forensic-enabled services is produced in order for the stakeholders and the software analysts to reason about the services that will finally be implemented in a foresicable way. This list illustrates the number of forensic constraints that are missing from a cloud service and the technologies that can be applied to in order for the service to become forensic-enabled. As mentioned earlier in the process, not all forensic constraints have applicability to a service since some of them may have already been implemented by the organization. A categorization can be produced to present the most costly cloud services; depending on how many constraints need to be implemented. According to this categorization, stakeholders can be aware of the cost of cloud forensic-enabled services and decide in accordance.

6.3.2 Evaluation-trace-back

The last step of the process concerns the assessment of the cloud services. After the hierarchy chart is produced an assessment takes place, where stakeholders evaluate if the chosen cloud services, which they intend to make forensic-enabled, can be implemented. If the evaluation is negative (for example the budget cannot support the implementation of the chosen cloud services), stakeholders may need to reconsider their strategy or may exclude a number of services of the implementation process. On the other hand, if the evaluation is positive (the budget allows the migration of more cloud services), stakeholders can go back to stage 2 and perform the cloud forensic requirements analysis to new services that they are willing to make forensic-enabled. This step is not mandatory and depends on the stakeholders' strategy.

6.4. Validation of the process

The research method employed in this paper is based in the Design Science Research Methodology (DSRM), created by [48]. In particular, it follows "*the six activities that make up the DSRM as a nominal sequence*" [49]. According to the Peffers et al. model, a problem statement was defined based on the identified gap in current research (activity 1), a literature review was conducted in order to find new ideas and solutions (activity 2), and a prototype was designed and developed to address the gap that existed in designing cloud forensic-enabled services (activity 3). The applicability of the proposed methodology was demonstrated through a case study involving a provider (activity 4), and its application on two different cloud services was evaluated (activity 5). Finally, a paper reporting the first stages of the methodology was published in the TrustBus 2017 international conference, while further publications of the methodology are underway (activity 6).

Based on the Peffers et al. methodology, Gregor and Hevner presented the DSR knowledge contribution framework [50], where the type of contribution is placed on four distinct quadrants. The four quadrants are the as follow:

- Invention, where new solutions for new problems are invented.
- Improvement, where new solutions for existing problems are developed.
- Exaptation, where existing solutions to new problems are extended.
- Routine design, where existing solutions to existing problems are applied.

The proposed process concerning the CFeS methodology belongs in the "Improvement" quadrant since the solutions that have been proposed in the context of digital forensics are evolved. On the other hand, this work moves a step forward by suggesting and providing new solutions in the cloud forensics. In this way, new boundaries are set to assist and define the specific field; hence, the proposed work also belongs in the "Invention" quadrant. The proposed CFeS methodology is a new idea and "little understanding of the problem context exists" [50]. It is an innovative work that defines new research questions and verifies the value of the solutions.

Table 7 illustrates the different DSRM activities, as they were applied in the context of this research along with the main results of each activity. The extra (fourth) column addresses the steps of the proposed process in the DSRM.

Table 7 DSRM applied to CFeS Methodology

DSRM activities	Activity description	Results	Addressed in the proposed process
Problem identification and motivation	There is a gap on development a methodology that can assist software engineers to design cloud services in a forensic sound manner	Literature review. Understanding the current solutions and their weaknesses	Define organizational strategy Identify cloud services (local and outsourced – Chapter 6.1)
Define the objectives of a solution	Design cloud services that are able to support cloud forensic investigations	Literature review. Knowledge of	Selection of cloud services (Chapter 6.2.1)

		emerging technologies, security and privacy requirements	
Design and development	Design and implementation of the CFeS Methodology: Cloud Forensic-enabled Services Methodology	Introduce forensic requirements (constraints). CFeS Methodology	Development of feature and activity diagrams (Chapter 6.2.1)
Demonstration	A case study demonstration using different services	Applying forensic requirements and CFeS to a real-world problem	Applicability of forensic constraints to cloud services (Chapter 7). Selection of forensic technologies in accordance to Table 8 (Chapter 7.2.3)
Evaluation	The CFeS Methodology met the project's objectives	Understanding the current solution and its weaknesses	The two services became forensic-enabled by signing contracts between the parties and performing specific actions (Chapter 7.3)
Communication	Published in the TrustBus 2017. To be published in a journal.	Understanding the forensic requirements and the need to design cloud forensic-enabled services	

7 Methodology Applicability: The University of the Aegean case study

For examining the applicability of the proposed methodology this was applied on a real case study, regarding the transformation of cloud services of the University of the Aegean (UoA) in order to make these services cloud forensic-enabled.

7. 1 Stage 1: Organizational analysis

The first stage of the proposed methodology is to identify and illustrate the organizational goals and the organization's cloud services. The main activity of the University is to introduce new approaches in higher education in Greece and worldwide and to promote regional development. Due to the fact that the UoA is located on 6 different islands in the Aegean Archipelagos, from its early days it has developed a modern network of IT infrastructures and services. The IT department constantly upgrades both its infrastructures and services and integrates the evolving technology of computer science. The UoA's objective is to bring the new technologies closer to education, research, and administration. A number of cloud services is provided to the academic community, such as e-mail services, web hosting, file storage, nextcloud etc. The UoA is equipped with a new technology data center (IBM) consisting of 22 blades (each one is equipped with 41,58 GHz processors, 256GB RAM) and it is managed by the VMware vSphere ESXi. It also uses IBM's Storwize V7000 for data storage with a capacity of 122TB. The UoA's goal for the following year is to provide the academic community with new and more efficient services by increasing its storage capacity. Both data center and storage are supported by a tape library, which takes backup of the systems on a daily basis. Databases, applications and software are accommodated in the Virtual Machines (VM) of the data center and the equipment is connected to a manageable IBM switch. The people responsible for managing the above equipment are the people who work in the central IT department of the UoA.

7.1.1 Define organizational strategy

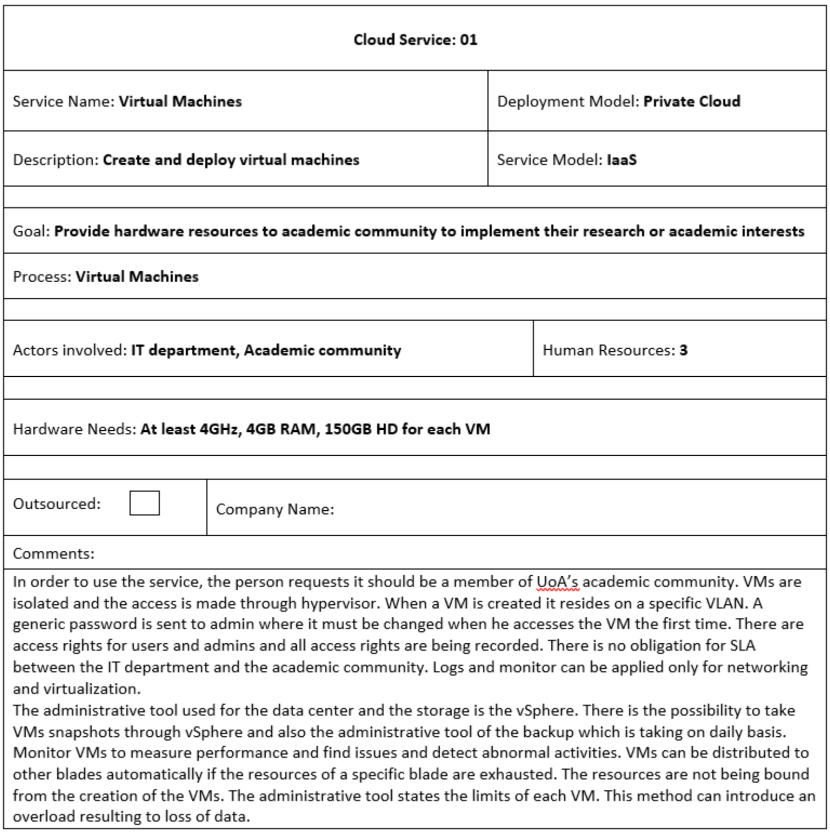
The main objective of the UoA's administration is to provide high quality research and education to the academic community. In order to achieve this (from a technical point of view) computer equipment needs to be updated on a regular basis and the services provided to the community need to be efficient and at the edge of technology. The infrastructure is constantly updated and the community is brought closer by using reliable services with fast connections. To support the venture, the UoA nodes (islands) are connected with each other through links with transmission speed of at least 1 Gbit (expandable to 10Gbit). To accomplish its objectives and bring new and reliable services to the academic community the UoA's strategy is to have a powerful IT department and infrastructures as described in the organizational analysis. Experienced personnel on information technology have been hired to manage the network and develop the services. Since the cloud is the technology used by most people nowadays, the UoA seeks and implements cloud services for the academic community. A report is produced describing UoA's IT architecture and the network connections between the nodes (islands). The report also includes the university's strategy related to the new information technologies. The actors involved in the process are the IT staff, the administration, teaching staff, students and some organizations that are using services for web hosting.

7.1.2 Identify and describe cloud services

The second step of the first activity is to identify the cloud services provided to the academic community. The cloud services related to the University are as follows:

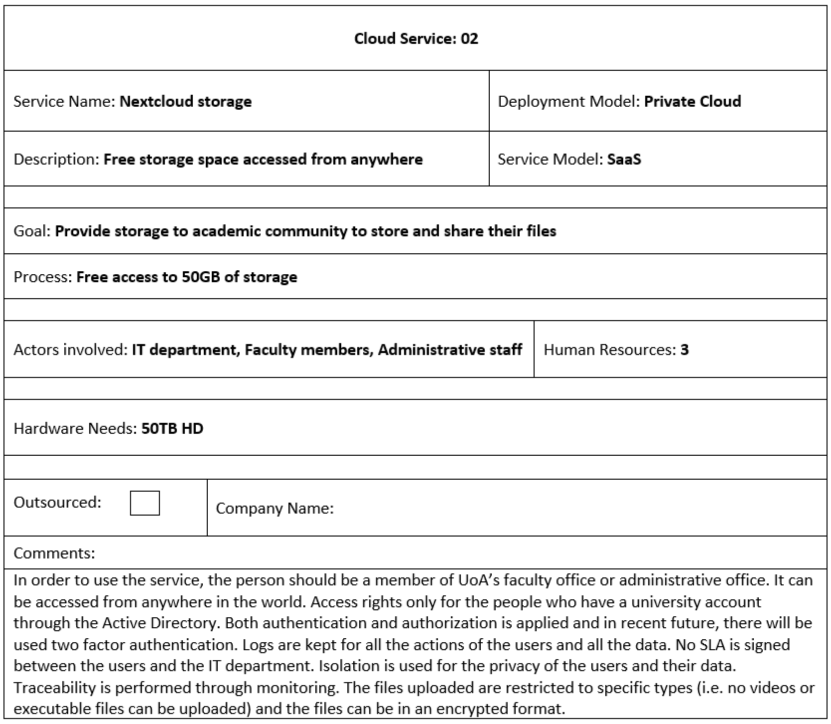
- Virtual Machines
- E-mail
- Web hosting
- File storage
- Nextcloud storage

For each service identified, a service cloud template is used to illustrate and describe all the aspects of the service. The output of this step is a cloud service description catalogue with all the necessary information. For the sake of the case study two specific services have been chosen to be thoroughly described, virtual machines and nextcloud storage. These two services will also be used to demonstrate the cloud forensic requirements analysis in stage 2. The results are highlighted in Figures 15 and 16.



Cloud Service: 01	
Service Name: **Virtual Machines**	Deployment Model: **Private Cloud**
Description: **Create and deploy virtual machines**	Service Model: **IaaS**
Goal: **Provide hardware resources to academic community to implement their research or academic interests**	
Process: **Virtual Machines**	
Actors involved: **IT department, Academic community**	Human Resources: **3**
Hardware Needs: **At least 4GHz, 4GB RAM, 150GB HD for each VM**	
Outsourced: □	Company Name:
Comments:	
In order to use the service, the person requests it should be a member of UoA's academic community. VMs are isolated and the access is made through hypervisor. When a VM is created it resides on a specific VLAN. A generic password is sent to admin where it must be changed when he accesses the VM the first time. There are access rights for users and admins and all access rights are being recorded. There is no obligation for SLA between the IT department and the academic community. Logs and monitor can be applied only for networking and virtualization. The administrative tool used for the data center and the storage is the vSphere. There is the possibility to take VMs snapshots through vSphere and also the administrative tool of the backup which is taking on daily basis. Monitor VMs to measure performance and find issues and detect abnormal activities. VMs can be distributed to other blades automatically if the resources of a specific blade are exhausted. The resources are not being bound from the creation of the VMs. The administrative tool states the limits of each VM. This method can introduce an overload resulting to loss of data.	

Fig. 15 Description catalogue for Virtual Machines service.



Cloud Service: 02

Service Name: **Nextcloud storage**	Deployment Model: **Private Cloud**
Description: **Free storage space accessed from anywhere**	Service Model: **SaaS**

Goal: **Provide storage to academic community to store and share their files**

Process: **Free access to 50GB of storage**

Actors involved: **IT department, Faculty members, Administrative staff**	Human Resources: **3**

Hardware Needs: **50TB HD**

Outsourced: □	Company Name:

Comments:

In order to use the service, the person should be a member of UoA's faculty office or administrative office. It can be accessed from anywhere in the world. Access rights only for the people who have a university account through the Active Directory. Both authentication and authorization is applied and in recent future, there will be used two factor authentication. Logs are kept for all the actions of the users and all the data. No SLA is signed between the users and the IT department. Isolation is used for the privacy of the users and their data. Traceability is performed through monitoring. The files uploaded are restricted to specific types (i.e. no videos or executable files can be uploaded) and the files can be in an encrypted format.

Fig. 16 Description catalogue for Nextcloud service.

7.1.3 Identify outsourced cloud services

The cloud services that the UoA provides to the academic community do not involve third providers' services due to the fact that they are implemented by the institution's own resources and its infrastructures are competent to do so. Thus, this step is not applicable to the whole process.

7. 2 Stage 2: Cloud forensic requirements analysis

In this stage, the University of the Aegean is willing to implement two services in order to make them forensic-enabled; virtual machines and nextcloud storage. These services are important to the university since critical data and applications are running and stored on them. Forensic constraints and technologies will be applied on these two services to realize the forensic requirements of each service.

7.2.1 Selection of cloud services

The first step of the second stage involves the selection of cloud services to be implemented as forensic-enabled. As mentioned earlier and according to the UoA's needs, the services that need to be implemented are the virtual machines and the nextcloud storage. For each service, an activity diagram is implemented as shown in Figures 17 and 18. The two activity diagrams describe all the activities and actions of the two selected services. All the activities derive from the service description catalogue.

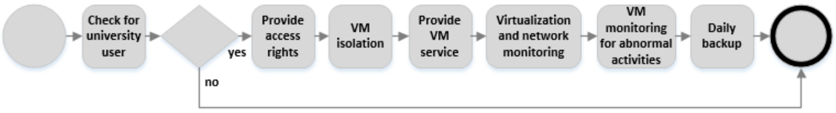

Check for university user
yes
no
Provide access rights
VM isolation
Provide VM service
Virtualization and network monitoring
VM monitoring for abnormal activities
Daily backup

Fig. 17 Activity diagram for Virtual Machine service.

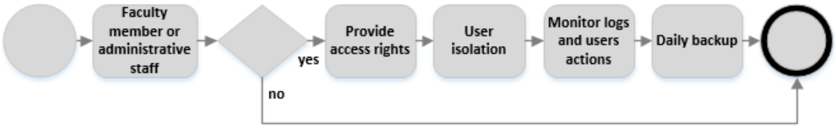

Faculty member or administrative staff
yes
no
Provide access rights
User isolation
Monitor logs and users actions
Daily backup

Fig. 18 Activity diagram for Nextcloud storage service.

7.2.2 Applicability of forensic constraints to cloud services

During this step, forensic constraints will be applied to the activity diagrams in order to make these two services cloud forensic-enabled. Taking under consideration the cloud service description catalogue, the activity diagrams and the feature diagrams that describe the tasks per constraint that need to be fulfilled, we can come to the conclusion that some forensic constraints are not satisfied. Once there is no SLA or contract signed between the two sides, forensic constraints such as accountability, internal disciplinary procedures and legal matters are not met. The IT department may know the identity of the user who owns the VM but they cannot be certain if the user is willing to hand its logs or even delete its data. On the other hand, VM snapshots are taken only if the IT administrator requests it.

Since the staff of the IT department is not obliged to sign any contract, accountability cannot be met. People working in the IT department are also not obliged to perform any surveillance or behavior policy; hence, there are issues with the internal disciplinary procedures. Legal matters, concerning the jurisdiction issues, are not applied, since the data center is not geographically distributed and the users are members of the UoA's academic community. However, as far as the contract agreements are concerned there are certain steps that need to be done such as signing contracts to ensure the terms of using cloud infrastructures. Finally, the access to the computer room where the data center and equipment are operating is restricted not only to the people responsible for the data center, but to all the personnel who is working in the IT. The transparency constraint is fulfilled only in the VM service since the UoA provides all three tasks in the specific service. As far as the Nextcloud service is concerned, the UoA does not provide any notification on policy violation, unless it is requested. Traceability is achieved in both cases through the monitoring system, access rights through the users' identification and isolation through the administrative tools and the methods used.

The analysis performed in the previous paragraphs concludes with the implementation of cloud forensic-enabled activity diagrams for each service. These two diagrams are shown in Figures 19 and 20. The black-colored boxes are the forensic constraints that need to be implemented so as the service to be cloud forensic-enabled.

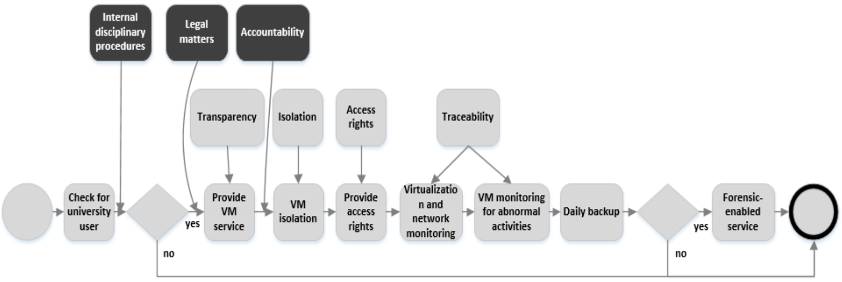


Fig. 19 Cloud forensic-enabled activity diagram for Virtual Machines service.

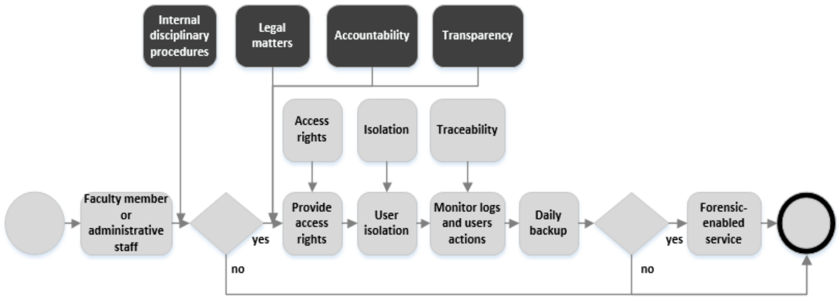


Fig. 20 Cloud forensic-enabled activity diagram for Nextcloud service.

As we can see from the two figures, the VM service needs three forensic constraints to be implemented so as to make it cloud forensic-enabled while the Nextcloud service needs four. For each forensic constraint that needs to be implemented the corresponding feature diagram is applied in order to identify the tasks that are not fulfilled by the constraint. Thus, in the VM service the feature diagrams of Internal Disciplinary procedures, Legal Matters and Accountability are considered as shown in Figure 21. Based on the VM activity diagram and the tasks of the feature diagram the missing tasks are easily identified (colored in black). Figure 22 presents the feature diagrams having identified the missing tasks regarding Next-cloud service. In Figures 23 and 24 the new activity diagrams for VM and Next cloud services are presented enhanced with the identified missing tasks respectively. For reasons of simplicity we present in detail only the missing tasks (as activities) of the missing forensic constraints in the new activity diagrams.

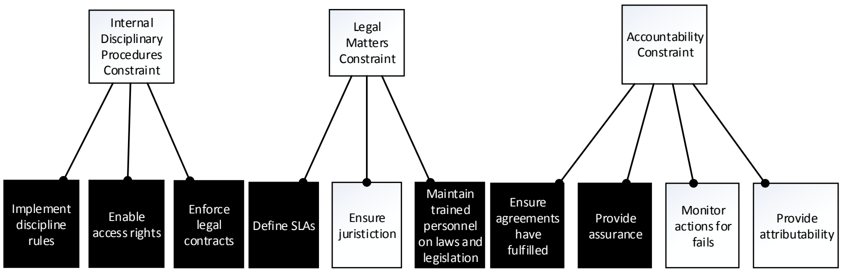

Internal Disciplinary Procedures Constraint
Legal Matters Constraint
Accountability Constraint
Implement discipline rules
Enable access rights
Enforce legal contracts
Define SLAs
Ensure juristiction
Maintain trained personnel on laws and legislation
Ensure agreements have fulfilled
Provide assurance
Monitor actions for fails
Provide attributability

Fig. 21 Identification of missing tasks for Virtual Machines service

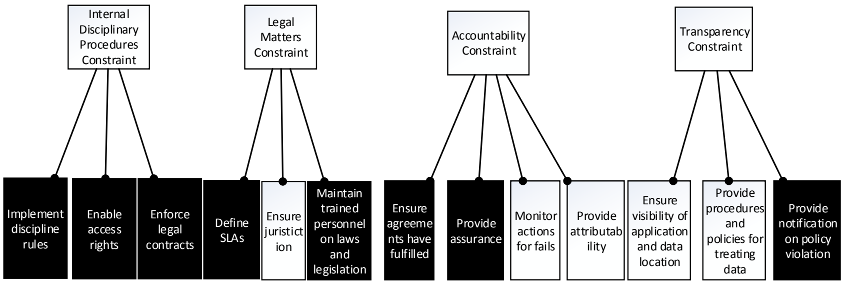

Internal Disciplinary Procedures Constraint
Legal Matters Constraint
Accountability Constraint
Transparency Constraint
Implement discipline rules
Enable access rights
Enforce legal contracts
Define SLAs
Ensure juristiction
Maintain trained personnel on laws and legislation
Ensure agreements have fulfilled
Provide assurance
Monitor actions for fails
Provide attributability
Ensure visibility of application and data location
Provide procedures and policies for treating data
Provide notification on policy violation

Fig. 22 Identification of missing tasks for Nextcloud service

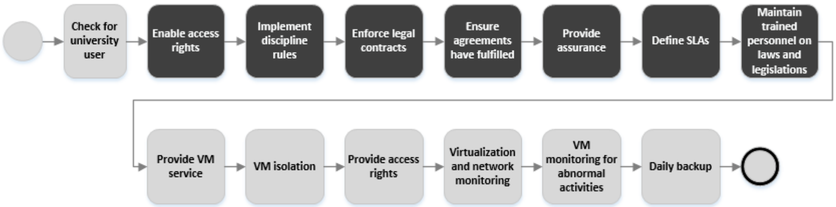

Check for university user
Enable access rights
Implement discipline rules
Enforce legal contracts
Ensure agreements have fulfilled
Provide assurance
Define SLAs
Maintain trained personnel on laws and legislations
Provide VM service
VM isolation
Provide access rights
Virtualization and network monitoring
VM monitoring for abnormal activities
Daily backup

Fig. 23 Activity diagram of missing tasks for Virtual Machines service

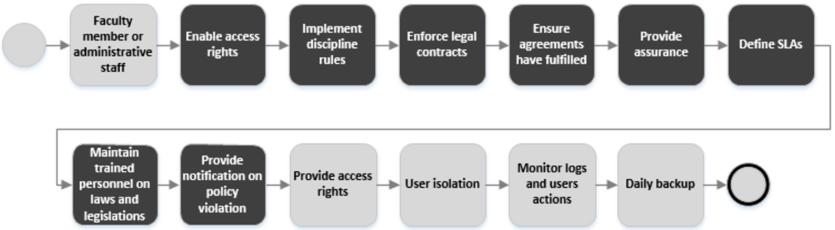

Faculty member or administrative staff
Enable access rights
Implement discipline rules
Enforce legal contracts
Ensure agreements have fulfilled
Provide assurance
Define SLAs
Maintain trained personnel on laws and legislations
Provide notification on policy violation
Provide access rights
User isolation
Monitor logs and users actions
Daily backup

Fig. 24 Activity diagram of missing tasks for Nextcloud service

7.2.3 Selection of technologies

This step involves the technologies identified from the literature that should be applied into forensic constraints. It is obvious from Figures 19 and 20 that the applicability of the technologies concerns only three forensic constraints for the first cloud service and only four for the second cloud service. These constraints have something in common; they all concentrate on SLAs to solve the issues among other techniques. Service Level Agreements are very important when a cloud provider is hiring its services and infrastructures to consumers and organizations.

In our case for the internal disciplinary procedures constraint, an SLA or a contract should be signed between the UoA and the IT staff responsible for the cloud services clearly stating the rules and the policies they should follow at all times. A mechanism should record and monitor their actions and a report should be sent to the IT administrator when an abnormal activity occurs. On the other hand, the accountability constraint should be solved again with an SLA. The UoA should assure that the consumers using its services are responsible and accountable for their actions and all the above should be written on the SLA. The IT's actions are not monitored nor recorded and this can lead to false assumptions. As far as we know, there is no vulnerability assessment or any penetration testing approach. Transparency constraint is partly fulfilled. Users have the freedom to handle and control their own computation and data according to their usage and they can also be certain that their data is securely backed-up and/or deleted according to their wishes. Again, since there is no SLA signed between them the boundaries are blurred. Finally, legal matters constraint related to jurisdictions issues and international law is not applied in our case study, since the data center is not geographically distributed and the users are members of the UoA's academic community. On the other hand, contractual terms, and constitutional issues are not satisfied. Thus, an SLA should also be signed between the two parties.

Earlier in the process, it was stated that some specific criteria (deployment model, service model and missing forensic-related constraints), need to be taken under consideration in order to select the technical solutions. Table 8 presents the criteria and suggested solutions for each cloud service in our case study.

Table 8 Criteria for selected solutions.

Service Name	Deployment Model	Service Model	Forensic Constraints	Suggested Solution
Virtual Machines	Private √	IaaS √	Internal disciplinary procedures	Sign robust SLA Enforce discipline rules Provide physical access rights to specific personnel
		PaaS □	Legal matters	Sign robust SLA Train personnel
	Public □	SaaS □	Accountability	Define SLA parameters Provide vulnerability assessment
Nextcloud	Private √	IaaS □	Internal disciplinary procedures	Sign robust SLA Enforce discipline rules Provide physical access rights to specific personnel
		PaaS □	Legal matters	Sign robust SLA Train personnel
	Public □	SaaS √	Accountability	Define SLA parameters Provide vulnerability assessment
			Transparency	Sign robust SLA

7. 3 Evaluation-Assessment

This is the last stage of the process and the administration of the UoA is called to decide whether the two cloud forensic-enabled services can be implemented or not.

7.3.1 Categorization of cloud forensic-enabled services

Based on the previous step “selection of technologies” and its applicability to the activity diagrams, the UoA’s administration realized that the cost of implementing both cloud services is within its budget. This arises from the fact that most of the technologies that resolve forensic constraints deal with Service Level Agreements and contracts between the two parties. There is no need to buy new equipment or to upgrade applications and software. Some penetrations tests that need to be performed are also within the UoA's budget.

7.3.2 Evaluation-trace-back

At this point, the UoA's administration decide to hold back the implementation of the rest of the services as cloud forensic-enabled due to the lack of financial resources. Even though its strategy is leaning towards the direction of implementing cloud forensic-enabled services, it is decided to proceed only as soon as the budget allows it.

8 Discussion

Information system designers have to face an important issue while designing cloud services. They have to design and implement cloud forensic-enabled services that could assist protective actors solve cloud-based cyber-crimes. After a thorough literature review, limited evidence of cloud-based forensic approaches was found, which do not support information systems developers as they focus only on the investigation. A gap in the field of cloud forensics exists since, to the best of our knowledge, no methodology exists for handling the design of cloud forensic-enabled services. In this paper, our proposed methodology aims to fill this gap by supporting the elicitation and modeling of forensic requirements. Specifically, it identifies seven forensic constraints that assist software engineers to implement cloud forensic-enabled services and it introduces a feature diagram for each constraint. Feature diagrams help software engineers by indicating a number of tasks needed to be fulfilled for the service to become forensicable.

The feature diagrams have been designed to support a generic approach in order to facilitate different environments in the future. This means that the tasks of the feature diagrams can be used to cloud-based services, or to traditional ones, such as web-services, or even services related to the future technologies. All the aforementioned seven cloud forensic constraints should be applied on a cloud service in order for it to be forensic-enabled. Applying the seven feature diagrams on the cloud service activity diagram of the methodology process can help software engineers to locate and identify the number of constraints that need to be implemented.

The methodology process presented in the paper is based on the concepts identified and presented in the conceptual model. It aims to identify an organization's strategy and needs, and use the identified feature diagrams in order to implement cloud forensic-enabled services. It thoroughly analyzes the structure of cloud services an organization provides to consumers and suggests the steps that need to be undertaken, as well as the technologies that need to be used to fulfill the organization's goals.

To assess the methodology's applicability, the proposed method has been performed in two different cloud services of the University of the Aegean. The results indicate that the methodology successfully identifies the organization's goals and forensic needs, and introduces technological activities and solutions based on the forensic requirements. These activities and solutions can guide software engineers to design and implement cloud forensic-enabled services. On the other hand, the applicability of the methodology allows organizations to have an overall picture of their cloud services and be more competitive, by recognizing their needs and costs for implementing cloud forensic-enabled services, compared to other organizations.

A limitation of the methodology is that the case study uses an organization that provides cloud services to consumers in a private cloud deployment model and does not have any dependencies on third parties, such as providers, brokers etc. Another limitation is that data is stored in data centers located in a specific geographical area (the islands of the Aegean Sea), thus the issue with different jurisdictions is not applied to the case study. Having created a methodology for implementing cloud forensic-enabled services for the needs of protective actors, the next phase is to extend methodology tests to other jurisdictions and include organizations with more dependencies on third parties.

This work provides a generic methodology that can assist software engineers in a way that they will be able to design and implement cloud forensic-enabled services with immediate impact on a cloud forensic investigation. The methodology is implemented in order to fill the gap of non-existing process models and methodologies in the area of cloud services in relation to cloud forensics. This methodology is raising the importance of the relation between a forensic-enabled system and an investigation process and how the latter is assisted when an incident occurs.

Another important aspect of this work, which is part of the proposed methodology, is the identification of a set of forensic constraints that apply in cloud forensics. The identification of the seven forensic constraints constitutes a first step towards the creation of a set of forensic requirements and a first effort to establish a new category of properties (concepts) in the requirements engineering. The constraints aim to follow a similar pattern with the security and privacy requirements.

9 Conclusions

The number of cloud services used by consumers increases rapidly. Meanwhile, the cloud-computing environment has attracted malicious actors who use vulnerable cloud services to perform illegal activities. Immediate attention should be given to the implementation of cloud services that will assist protective actors to investigate incidents in a forensically sound manner. Our aim is to present a methodology that will assist designers to build cloud forensic-enabled services. This has been achieved with the introduction of a methodology that supports cloud services by implementing a number of steps in order to make the services cloud forensic-enabled. Specifically, we identified and proposed seven forensic constraints that should all be included in the design and implementation of any cloud service. For each forensic constraint, a feature diagram is introduced. A conceptual model is also presented based on the concepts and the forensic constraints requirements identified. The conceptual model is based both on the concepts that make a system forensic-enabled and on the concepts that form a cloud forensic investigation process. To conclude the methodology, a process has been developed based on the concepts identified and presented in the conceptual model. The process illustrates the stages and the activities that need to be followed to produce cloud forensic-enabled services. Finally, a case study is presented to test and assess the applicability of the methodology.

Future steps include a more thorough and precise evaluation of the methodology involving more cloud services and different providers that use third parties and multiple jurisdictions. The idea is to perform an evaluation of the methodology by independent organizations or practitioners in order to make the proposed methodology a standard for the community, the cloud providers and the cloud forensic investigation.

References

1. Skyhigh (2016) Cloud Adoption & Risk Report Q4 2016. Skyhigh. p 33
2. Martini B, Choo K-KR (2014) Distributed filesystem forensics: XtreemFS as a case study. Digital Investigation 11(4):295-313
3. Wilshusen GC (2016) Federal Information Security: Actions Needed to Address Challenges. U.S. Government Accountability Office. Washington, DC, USA, p 17
4. Simou S, Kalloniatis C, Mouratidis H, Gritzalis S (2016) Towards a Model-Based Framework for Forensic-Enabled Cloud Information Systems. In: Katsikas S, Lambrinoudakis C, Furnell S (eds) Proceedings of the Trust, Privacy and Security in Digital Business: 13th International Conference, TrustBus 2016. Porto, Portugal. Springer International Publishing: Switzerland, pp 35-47
5. McKemmish R (1999) What is forensic computing? Trends and issues in crime and criminal justice. Canberra, Australia: Australian Institute of Criminology vol 118:1-6
6. Palmer G (2001) A road map for digital forensic research. Technical Rreport from the First Digital Forensics Research Workshop (DFRWS). In: Proceedings of the First Digital Forensic Research Workshop. Utica, New York, USA, pp 1-48
7. U.S. Department of Justice (2001) Electronic Crime Scene Investigation: A Guide for First Responders. In: NIJ Research Report, NCJ 187736. Washington, p 96
8. Reith M, Carr C, Gunsch G (2002) An examination of digital forensic models. International Journal of Digital Evidence. 1(3):1-12.

9. Carrier B, Spafford EH (2003) Getting physical with the digital investigation process. International Journal of digital evidence. 2(2):1-20

10. Baryamureeba V, Tushabe F (2004) The enhanced digital investigation process model. In: Proceedings of the Fourth Digital Forensic Research Workshop (DFRWS). Baltimore, MD, USA

11. Ciardhuáin SÓ (2004) An extended model of cybercrime investigations. International Journal of Digital Evidence. 3(1):1-22

12. Selamat SR, Yusof R, Sahib S (2008) Mapping process of digital forensic investigation framework. International Journal of Computer Science and Network Security. 8(10):163-169

13. Beebe NL, Clark JG (2005) A hierarchical, objectives-based framework for the digital investigations process. Digital Investigation: The International Journal of Digital Forensics & Incident Response. 2(2):147-167

14. Kent K, Chevalier S, Grance T, Dang H (2006) Guide to integrating forensic techniques into incident response. NIST Special Publication. SP 800-86: p 121

15. von Solms S, Louwrens C, Reekie C, Grobler T (2006) A Control Framework for Digital Forensics. In: Olivier MS, Shenoi S (eds) Proceedings of the IFIP international Conference on Digital Forensics, National Center for Forensic Science. Advances in Digital Forensics II. Orlando, Florida, Springer New York: Boston, MA, pp 343-355

16. Cohen FB (2010) Fundamentals of digital forensic evidence. In: Stavroulakis P, Stamp M (eds) Handbook of Information and Communication Security. Springer Berlin Heidelberg, pp 789-808

17. Agarwal A, Gupta M, Gupta S, Gupta SC (2011) Systematic digital forensic investigation model. International Journal of Computer Science and Security (IJCSS). 5(1):118-131

18. Valjarevic A, Venter HS (2012) Harmonised digital forensic investigation process model. In: Proceedings of the 2012 Information Security for South Africa (ISSA). Johannesburg, South Africa, pp 1-10

19. Guo H, Jin B, Shang T (2012) Forensic investigations in cloud environments. In: Proceedings of the 2012 International Conference on Computer Science and Information Processing (CSIP). Xi'an, Shaanxi, pp 248-251

20. Chen G, Du Y, Qin P, Du J (2012) Suggestions to digital forensics in Cloud computing ERA. In: 3rd IEEE International Conference on Network Infrastructure and Digital Content (IC-NIDC). Beijing, China, pp 540-544

21. Martini B, Choo K-KR (2012) An integrated conceptual digital forensic framework for cloud computing. Digital Investigation 9(2):71-80

22. Ruan K, Carthy J (2012) Cloud Forensic Maturity Model. In: Rogers M, Seigfried-Spellar KC (eds) Proceedings of the 4th International Conference on Digital Forensics and Cyber Crime (ICDF2C). Springer: Berlin, Heidelberg, pp 22-41

23. Adams R (2013) The Emergence of Cloud Storage and the Need for a New Digital Forensic Process Model. In: Ruan K (ed) Cybercrime and Cloud

Forensics: Applications for Investigation Processes. IGI Global: Hershey, PA, USA, pp 79-104

24. Kohn MD, Eloff MM, Eloff JH (2013) Integrated digital forensic process model. Computers & Security. vol 38:103-115

25. Zawoad S, Hasan R, Skjellum A (2015) OCF: An Open Cloud Forensics Model for Reliable Digital Forensics. In: IEEE 8th International Conference on Cloud Computing (CLOUD). New York City, NY, pp 437-444

26. Simou S, Kalloniatis C, Gritzalis S, Mouratidis H (2016) A survey on cloud forensics challenges and solutions. Security and Communication Networks. 9(18):6285-6314

27. Simou S, Kalloniatis C, Kavakli E, Gritzalis S (2014) Cloud Forensics: Identifying the Major Issues and Challenges. In: Jarke M, Mylopoulos J, Quix C, Rolland C, Manolopoulos Y, Mouratidis H, Horkoff J (eds) Proceedings of the 26th International Conference on Advanced Information Systems Engineering (CAiSE). Thessaloniki, Greece, Springer International Publishing: Cham, pp 271-284

28. Kalloniatis C, Mouratidis H, Vassilis M, Islam S, Gritzalis S, Kavakli E (2014) Towards the design of secure and privacy-oriented information systems in the cloud: Identifying the major concepts. Computer Standards & Interfaces. 36(4):759-775

29. Zawoad S, Hasan R (2015) FECloud: A Trustworthy Forensics-Enabled Cloud Architecture. In: Peterson G, Shenoi S (eds) Advances in Digital Forensics XI. Springer International Publishing, pp 271-285

30. Liu F, Tong J, Mao J, Bohn R, Messina J, Badger L, Leaf D (2011) NIST cloud computing reference architecture. In: NIST special publication. National Institute of Standards and Technology, SP 500-292, p 35

31. Catteddu D, Felici D, Hogben G, Holcroft A, Kosta E, Leenes R, Millard C, Niezen M, Nunez D, Papanikolaou N (2013) Towards a model of accountability for cloud computing services. In: Proceedings of the DIMACS/BIC/A4Cloud/CSA International Workshop on Trustworthiness, Accountability and Forensics in the Cloud (TAFC). Malaga, Spain

32. Cloud Accountability Project (2016) Accountability in the Cloud - Coceptual Framework. [cited 2018 February 18]; Available from: http://a4cloud.eu/about.html

33. Newcombe L (2012) Securing Cloud Services: A pragmatic approach to security architecture in the Cloud. IT Governance Publishing

34. NIST (2013) NIST cloud computing security reference architecture. In: Working document, Draft SP 500-299. National Institute of Standards and Technology, p 204

35. Ruan K, Carthy J, Kechadi T, Crosbie M (2011) Cloud Forensics. In: Peterson G, Shenoi S (eds) Proceedings of the 7th IFIP WG 11.9 International Conference on Digital Forensics. Advances in Digital Forensics VII. Springer Berlin Heidelberg. pp 35-46

36. Chang C, Ramachandran M (2016) Towards Achieving Data Security with the Cloud Computing Adoption Framework. Transactions on Services Computing. 9(1):138-151

37. Kalloniatis C, Kavakli E, Gritzalis S (2008) Addressing privacy requirements in system design: the PriS method. Requirements Engineering. 13(3):241-255

38. Shei S, Kalloniatis C, Mouratidis H, Delaney A (2016) Modelling Secure Cloud Computing Systems from a Security Requirements Perspective. In: Katsikas S, Lambrinoudakis C, Furnell S (eds) Proceedings of the Trust, Privacy and Security in Digital Business: 13th International Conference, TrustBus 2016. Porto, Portugal. Springer International Publishing: Switzerland, pp 48-62

39. Simou S, Kalloniatis C, Kavakli E, Gritzalis S (2014) Cloud Forensics Solutions: A Review. In: Iliadis L, Papazoglou M, Pohl K (eds) Proceedings of the 4th International Workshop on Information Systems Security Engineering (WISSE). Advanced Information Systems Engineering Workshops: CAiSE 2014. Springer International Publishing: Cham, pp 299-309

40. Czarnecki K, Eisenecker UW (2000) Generative Programming: Methods, Tools, and Applications. 1st ed. Addison-Wesley

41. Šípka M (2005) Exploring the commonality in feature modeling notations. In: Bielikova M (ed) Proceedings of IIT. SRC. pp 139-144

42. Kavakli E, Kalloniatis C, Loucopoulos P, Gritzalis S (2006) Incorporating privacy requirements into the system design process: the PriS conceptual framework. Internet research. 16(2):140-158

43. ENISA (2013) Cloud Computing Incident Reporting: Framework for reporting about major cloud security incidents. p 38

44. Beebe N, Clark J (2005) Dealing with Terabyte Data Sets in Digital Investigations. In: Pollitt M, Shenoi S (eds) Proceedings of the IFIP international Conference on Digital Forensics, National Center for Forensic Science. Advances in Digital Forensics. Orlando, Florida, Springer New York: Boston, MA, pp 3-16

45. Grispos G, Storer T, Glisson WB (2012) Calm Before the Storm: The Challenges of Cloud Computing in Digital Forensics. International Journal of Digital Crime and Forensics (IJDCF). 4(2):28-48

46. Kokolakis S, Demopoulos AJ, Kiountouzis EA (2000) The use of business process modelling in information systems security analysis and design. Information Management & Computer Security. 8(3):107-116

47. Alotaibi Y, Liu F (2014) A novel secure business process modeling approach and its impact on business performance. Information Sciences. 277(Supplement C):375-395

48. Peffers K, Tuunanen T, Rothenberger MA, Chatterjee S (2007) A Design Science Research Methodology for Information Systems Research. Journal of Management Information Systems. 24(3):45-77

49. Geerts GL (2011) A design science research methodology and its application to accounting information systems research. International Journal of Accounting Information Systems. 12(2):142-151

50. Gregor S, Hevner AR (2013) Positioning and presenting design science research for maximum impact. MIS Quarterly. 37(2):337-356