FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Security Assurance Assessment for Multi-layered and Multi-tenant Hybrid Clouds

PhD THESIS

submitted in partial fulfillment of the requirements for the degree of

Doctor of Technical Sciences

by

mag.ing.el.techn.inf. Aleksandar Hudic
Registration Number 1029196

to the Faculty of Informatics

at the TU Wien

Advisor: Privatdoz. Dipl.-Ing. Mag.rer.soc.oec. Dr.techn. Edgar R. Weippl

The dissertation has been reviewed by:

Assistant Prof. Doc. Dr. sc. Stjepan Groš	Univ. Prof. Priv.-Doz. DI Dr. Rene Mayrhofer

Vienna, 1st October, 2017

Aleksandar Hudic

Declaration of Authorship

mag.ing.el.techn.inf. Aleksandar Hudic
Sechshauser Straße 31/552, 1150 Wien

I hereby declare that I have written this Doctoral Thesis independently, that I have completely specified the utilized sources and resources and that I have definitely marked all parts of the work - including tables, maps and figures - which belong to other works or to the internet, literally or extracted, by referencing the source as borrowed.

Vienna, 1. Oktober 2017

Aleksandar Hudic

Acknowledgements

I am sincerely grateful to my advisor Dr. Edgar Weippl, first of all for giving me the opportunity to pursuit the PhD and who supervised and offered guidance through my PhD and has always supported me in my own research ideas - in addition to patience and tolerance towards my ambiguous ideas.

I would also like to give a special thanks to my dear friend, colleague, and supervisor Dr. Paul Smith who, even in the busiest times managed to offer extensive support and feedback, as well as intensive discussion of ideas that helped me a lot.

Furthermore, I would like to give a special thanks to the people with whom I've worked in the Austrian Institute of Technology.

Beyond AIT, I would like to thank my friends Petar, Hrvoje, Ivan, and Ante that were always supporting and encouraging me during the busiest and challenging times.

I would like to express my deepest thanks to my family too. This includes my parents, Veljko and Milka, my sister, Mirela, my grandparents, Petar, Bosiljka, Ladislav, Nada, and my dearest uncle Bosko.

Finally, I would like to give a special thanks to my Sonja, who was my most faithful motivator that encouraged me continuously without any doubts during this wonderful journey.

Abstract

This thesis is based on five publications in the area of security assurance for hybrid cloud environments that were published at distinguished conferences and journals of IEEE and Elsevier. Cloud computing, as an ongoing and evolving research field, received an increasing attention in last several years. Despite the tremendous benefits, the cloud paradigm is confronted with challenges that arise on daily basis in its ecosystem, especially with the hybrid cloud model. Even though, hybrid cloud model introduced more control it also entailed more complexity. Meanwhile, the old challenges with regards to security, privacy, and especially transparency haven't been comprehended and therefore lack behind the technological momentum. Outsourcing security sensitive information and services in a cloud has been a main obstacle due to the immense lack of transparency. The hybrid cloud solution model attempts to mitigate the control problem by hosting sensitive information on private infrastructure placing the rest off-premise. Nevertheless, the transparency of cloud environments still remains unresolved. Hence, this thesis addresses these challenges, the transparency of cloud providers in particular, by validating security objectives across multi-layered hybrid cloud solutions to ensure that there are supporting security measures in place. The thesis focused its attentions and concerns towards the security and transparency of security critical services when hosting them in hybrid multi-layered and multi-tenant environments (i.e., hybrid clouds, industry 4.0).

To address the complexity of hybrid cloud environment that is often segmented across multiple layers and owned by multiple stakeholders, this thesis proposes a composite multi-layer reference architecture model. The model is used to support design, implementation and demonstration of a unique security assurance framework. The proposed framework demonstrates efficient acquisition and assessment of security related information across multidimensional critical infrastructure systems, i.e., different levels of abstractions or viewpoints. Furthermore, to implement an efficient data acquisition model for deploying critical infrastructure services to cloud environments the security related challenges, objectives and requirements were addressed. The identified challenges highlight the shortcomings of cloud providers when it comes to supporting transparency especially with regards to the hybrid cloud solution models. To overcome this gap, this thesis proposes a novel approach for holistic security assurance assessment that addresses the interdependencies between both individual components and abstraction levels in hybrid cloud environments. The approach offers the ability to assess each individual component of a hybrid infrastructure, regardless if it is a physical server, virtual container,

embedded system with limited resources or a high level service, in a structural manner by including all its interdependencies. The flexibility of the approach lies in the composite structural design of the security assurance assessment framework that adheres to the Common Criteria and enhances it to achieve a higher level of granularity when assessing services. Most importantly, unlike standard approaches for security assessment such as certification or auditing, the proposed model offers continuous security assessment ability and therefore nearly real time assessment of hybrid cloud environments where we can have competitive cloud provider that delivers one single service. The security assurance assessment model prevents the exposure of internal security sensitive information of a cloud provider via its novel security assurance assessment model that operates on abstracted security information sets. Furthermore, a comprehensive life-cycle model was proposed to integrated security through design, development, deployment, migration and maintenance of services in line with standards, regulative compliance, and best practices. The proposed model integrates iterative security requirements engineering from high level objectives to security properties used for security validation entities through both development and production phase of cloud services. Furthermore, the life-cycle aligns and integrates the security assurance assessment model, while at the same time considering the security requirements, enhancing transparency in production phase and implementing high-scale security automation.

Kurzfassung

Diese Arbeit basiert auf fünf Publikationen, die zum Bereich Security Assurance für hybride Cloud Umgebungen gezählt werden können, und bei Konferenzen und Journalen von IEEE und Elsevier präsentiert bzw. publiziert wurden. Cloud Computing ist ein aktuelles Forschungsfeld, welches eine vermehrte Aufmerksamkeit in den letzten Jahren erhalten hat. Neue Herausforderungen entstanden in Cloud Umgebungen während der täglichen Arbeit, besonders mit dem Aufkommen von neuen hybriden Cloud Modellen. Währenddessen sind die alten Herausforderungen im Hinblick auf Security, Privacy und besonders Transparenz nicht oder nur sehr unzureichend adressiert worden, um mit dem technologischen Fortschritt mithalten zu können, welche durch das Cloud Paradigma entstanden sind. Das Teilen von sicherheitsrelevanten Information in einer Cloud Umgebung ist das größte Hindernis aufgrund des Fehlens von Transparenz. Hybrid Cloud Modell Versucht dieses Hindernis zu adressieren, indem es die sensitive Daten lokal behandelt. Trotzdem Transparenz bleibt ungelöst. Daher befasst sich diese Doktorarbeit mit diesen Herausforderungen, insbesonders im Hinblick auf die Transparenz der Cloud Provider, die unterstützende Maßnahmen für die gegebenen Sicherheitsziele erhalten. Die Arbeit befasst sich vor allem mit Sicherheit und Transparenz in Bezug auf sicherheitskritische Dienste, vor allem auf in hybriden multi-layered und multi-tenant Umgebungen.

Um die Komplexität der hybriden Cloud Umgebungen zu verstehen, die aus mehreren Schichten zusammengesetzt und im Besitz von mehreren Beteiligten sein können, zeigen wir eine zusammengesetzte multi-layer Referenzarchitekturmodell. Dieses Modell abstrahiert einzelne verschiedene abstrakte Ebenen quer durch die Cloud (zum Beispiel Benutzer, Dienstleistungen, Mieter und physisch). Das Hauptziel dieses Modells ist die Beobachtung von mehrdimensionalen kritischen Infrastruktur auf einzelnen Ebenen aus verschiedenen Blickwinkeln, und zwar für multi-provider und multi-tenant und für verschiedene Interessensgruppen. Darüber hinaus analysieren wir die Herausforderungen, Ziele und Anforderungen für bereitgestellte kritische Infrastrukturdienste in bestehenden Cloud Umgebungen in Bezug auf Transparenz und Sicherheit. Die Herausforderungen, die wir identifiziert haben, unterstreichen die Unzulänglichkeit der Cloud-Anbieter Transparenz zu unterstützen, insbesondere im Hinblick auf hybride Cloud-Lösungen. Zur Überwindung dieser Lücke schlägt diese Arbeit ein neuartiges Modell für ein ganzheitliches Security Assurance Assessment vor, welche die Abhängigkeiten der einzelnen Komponenten und der Abstraktionsebenen in hybriden Cloud Umgebungen adressiert. Dieser Ansatz bietet die Möglichkeit, jede einzelne Komponente einer Cloud Infrastruktur, unabhängig

davon, ob es ein physischer Server, ein virtuellen Container oder ein High-Level Service ist, in strukturierter Art und Weise, einschließlich aller Abhängigkeiten, zu prüfen. Die Flexibilität des Ansatzes liegt im strukturiertem Aufbau des Security Assurance Assessment Framework, der die Common Criteria beinhaltet und ermöglicht eine höhere Granularität zu erzielen, um Services zu bewerten. Am wichtigsten ist, im Gegensatz zu Standardansätzen für die Risikobewertung, wie Zertifizierungen oder Revision, dass unser Modell eine kontinuierliche Risikobewertung von hybriden Cloud Umgebungen bietet, in denen wir wettbewerbsfähige Cloud-Anbieter haben, die einen einzigen Service liefern. Schließlich verhindert unser Security Assurance Assessment Modell die Offenlegung von internen sicherheitssensibler Informationen eines Cloud-Anbieters durch das neue Security Assurance Assessment Modell, das auf einem abstrahierten Set von Sicherheitsinformationen arbeitet. Darüber hinaus schlagen wir einen umfassenden Life-Cycle für Design, Entwicklung und Bereitstellung von sicheren Cloud-Diensten im Einklang mit Standards, regulative Compliance und Best Practices vor. Zusätzlich integriert der vorgeschlagene Life-Cycle iterativ Sicherheitsanforderungen von High-Level-Zielen bis zu Sicherheitseigenschaften für Sicherheitsvalidierung, sowohl während der Entwicklung als auch während der Produktionsphase von Cloud-Services. Der Life-Cycle ist ausgerichtet auf und integriert gleichzeitig das Security Assurance Assessment Modell, indem Sicherheitsanforderungen in der letzten Produktionsphase unterstützt werden um Transparenz zu verbessern.

Contents

CHAPTER 1

Introduction

Over the course of time, the Information and Communications Technology (ICT) revolution converged a variety of technological concepts, methodologies and trends, like virtualization [APST05], service oriented design [PvdH06], distributed computing [TVS02], into a novel model for services and resources utilization. Consequently, service utilization overcame both physical and administrative domain boundaries of a single computer system and transformed traditional service utilization towards a more resilient, interconnected and global computation model supported with high scalability, performance, and availability characteristics.

The interconnection of distinct computer systems into a coherent computation and availability infrastructure, supported with high-performance microprocessors, high-speed networks, and high performance distributed computing emerged the first distributed computer system model nowadays familiar as *Cluster computing* [YBP+06]. An example of such cluster computing concept was founded as a part of Beowulf [SSB+95], Berkeley [Pat94], and HPVM [CPL+97] academic projects. The concept demonstrated significant advancements in terms of availability, scalability and performance over the traditional computing systems. Furthermore, cluster computing found a widespread appliance in industry afterwards (e.g., Microsoft Cluster Server Service[1], data analysis [ZCF+10, HYC16]). The next technological evolution step enhanced the cluster computing model towards a more heterogeneous and non-interactive computational model by dispersing computation and service across multiple distinct locations, *Grid computing* [FIvL+99, FK99]. One of the main objectives of the grid computing model is to perform orchestrated resource sharing and problem solving in dynamic, multi-institutional virtual organizations. The resource orchestration model directly accesses computers, software, data, and other resources, as it is required by a range of collaborative problem-solving and resource-brokering strategies emerging in industry, science, and

[1]Microsoft Cluster Service (MSCS) `https://msdn.microsoft.com/en-us/library/ms952401.aspx`,(last visited December 11, 2016)

engineering [FKT01, FK99]. The shortcoming of the grid computing concept was the lack of interactive computation that was essentially resolved by the next generation technological successor, the *Cloud Computing*. National Institute of Standards and Technology (NIST) defined cloud computing as a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction [MG11]. Most importantly, NIST also introduced a cloud reference architecture model [BML+11] that defines five essential cloud computing actors, (cloud consumer, cloud provider, cloud carrier, cloud auditor and cloud broker) their roles, and mutual relationships. Cloud Computing can be also seen as a successor that converged from distinct technological concepts like virtualization, service and resource distribution, grid and cluster computing, enterprise service management, into a revolutionized service delivery model. The collaboration and orchestration of services and resources in cloud computing model are the advancements that permanently transformed the landscape of ICT services and at the same time offered considerable economic benefits. Essentially, the cloud computing paradigm became a prominent solution for offering enhanced ubiquitous services with high scalability and availability characteristics. This significantly enhanced the abilities of traditional service deployment beyond single administrative and geographical domain towards a globally interconnected computing model. In the beginning of 2008, Gartner research[2], indicated the Cloud computing paradigm as one of the most promising technologies which became, notably, one of the most disruptive and prominent technologies nowadays. Cloud computing essentially evolved to a next stage service utilization model, hybrid cloud [Lin16], that became even more interconnected with a multidimensional inter-domain computation model[3]. The step towards hybrid is based on the assumption that companies have specific architectures, application landscapes, and data that they will not move to public clouds [You16]. Regardless of complexity, the hybrid cloud models provide the best effort from both private and public cloud worlds, by combining the economies and efficiencies of public cloud computing with the security and control of private cloud computing [Lin16].

The evident evolution of ICT systems and services towards an interconnected multidimensional and inter-discipline ICT systems model, i.e. hybrid infrastructure models, are unfortunately opposed by monitoring, compliance, auditing, integrity challenges with respect to security and privacy [SK11, KPR09, ZL12, NCAW14, YK13, Lin16]. Although security and privacy challenges emerged as a consequence of technological evolution of service oriented model, they did not evolve along, and therefore leaving a significant gap. Consequently, cloud computing faced major security breaches. Some of the most recent major information security leaks, e.g. database of 191 million U.S.

[2]Gartner Top Ten Disruptive Technologies `http://www.gartner.com/newsroom/id/739613`,(last visited December 11, 2016)

[3]Hype Cycle for Emerging Technologies `http://www.gartner.com/newsroom/id/3114217`,(last visited December 11, 2016)

voters exposed[4], storing passwords without salt[5,6], server misconfiguration[7], Baracuda networks default password[8] happened due to the misconfiguration or lack of proper security protection. Even if the providers claim that users' information is protected there is no guarantee that at certain point of time a part of the system, as it is shown in the above mentioned security incidents, is improperly configured [TJA10]. The ICT environments are commonly put to an extensive security assessments as standard best practice by performing audits, certifications, and continuous monitoring to protect their assets. Nonetheless, no matter how detailed or comprehensive these approaches are, they still fail to detect issues continuously, but rather detect them at the end of a certain time window, i.e., that happens in between two audits, assessments or certification processes. A very distinctive challenge that occurs when performing security assessments, especially in multidimensional infrastructures, is the time and effort required to perform security analysis. Security assessments also commonly require extensive human intervention during the process. Unfortunately, human involvement in such complex processes is inevitable and often prone to errors. As such, human intervention is more a liability that due to the lack of automated assessment process support has to be properly addressed.

The automation processes offered by standard infrastructure monitoring solutions are commonly focused to derive only infrastructure related information with respect to functionality, performance and availability, and less related to security. Thus, security assessment can become especially challenging when it comes to inter-domain multidimensional systems such as hybrid infrastructure environments. Commonly, services in such models are spanned beyond a single administrative and geographical domain. An example of such administrative and geographical dispersion is Dropbox [DMM+12] which uses Amazon S3[9] storage for storing user' data and therefore independent layers are being owned by different stakeholders. In the case of Dropbox [DMM+12] the users were assured that their data was securely encrypted. However, this was not technically clarified to show what security measures were actually in place and at which point. Was the customer data encrypted in transition, at rest, or both? The problem was that in this particular case the underlying cloud infrastructure provider, Amazon, unfortunately did not offer any kind of encryption for the data at rest at that point of time which essentially left the users' data completely unprotected. The Dropbox system architecture model is a very convenient example of a multi-layered distributed service that spans across

[4] http://uk.reuters.com/article/us-usa-voters-breach-idUKKBN0UB1E020151229,(last visited December 1., 2016)

[5]MySpace http://www.refinery29.com/2016/06/112737/myspace-hack,(last visited December 11, 2016)

[6]Linkedin http://thehackernews.com/2016/05/linkedin-account-hack.html,(last visited December 11, 2016)

[7]MacKeeper http://krebsonsecurity.com/2015/12/13-million-mackeeper-users-exposed/,(last visited December 11, 2016)

[8]Baracuda Networks default password vulnerability http://www.dailymail.co.uk/news/article-3055281/Global-vendor-payment-devices-uses-default-password-not-changed-1990.html,(last visited December 11, 2016)

[9]Amazon S3 storage https://aws.amazon.com/documentation/s3/,(last visited December 11, 2016)

multiple diverse cloud providers. In case of such multi-stakeholder scenario detailed security assessment would be required from each of the stakeholder to expose internal sensitive information to perform detailed holistic security analysis. To address the above mentioned challenges with regards to security assessment, Common Criteria [Her02] system assessment model based on ISO15408-3 [fSC08a] is imposed as most promising approach nowadays for systematic security assessment due to its holistic security analysis characteristics. Furthermore, the common criteria abstracts the security assessment result in a form of assurance levels and therefore avoids exposure of internal infrastructure-sensitive information. Nevertheless, the structured and systematic approach of Common Criteria is unfortunately designed only to indicate type of tests that have been performed on a system (functional, structural, methodical, semi-formal and formal) that eventually determines the level of assurance.

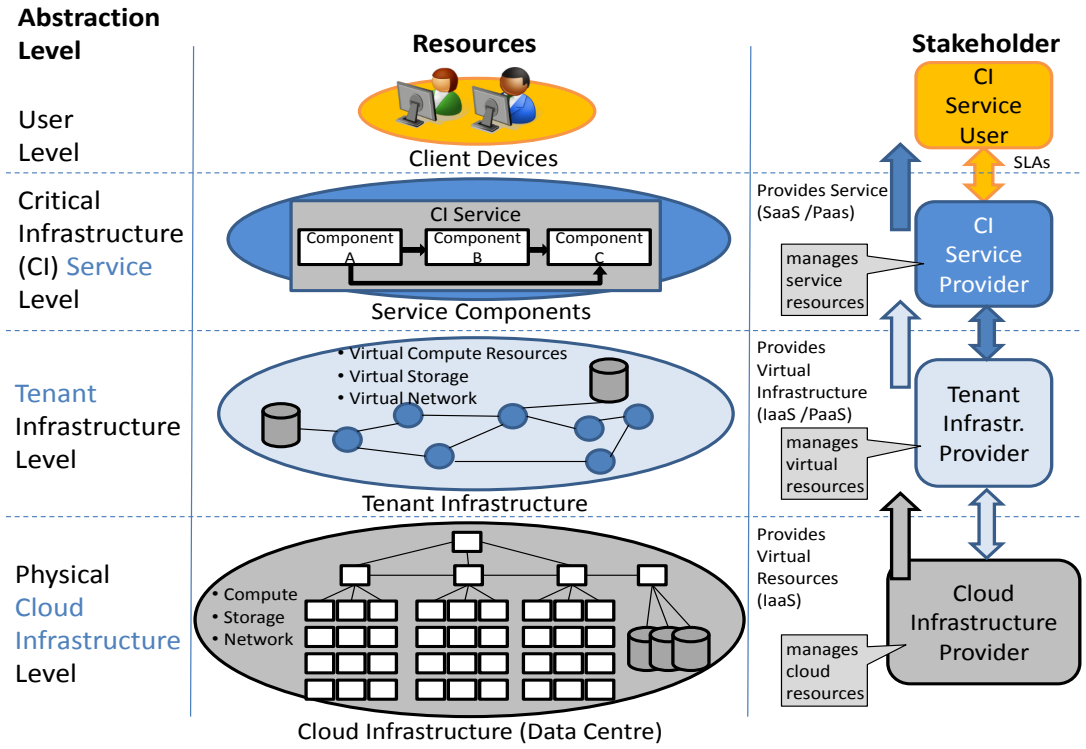

Figure 1.1: Reference architecture model of SEcure Cloud computing for CRitical infrastructure IT (SECCRIT) research project [BFH+15]

The NIST architectural framework refers to a multi-layered model such as Dropbox without the multi-stakeholder part though. Therefore, for the purpose of this thesis and the research project that this thesis was a part of, SEcure Cloud computing for CRitical infrastructure IT (SECCRIT), the reference architecture model was defined, shown in Figure 1.1. The reference architecture model aims for a more precise role distinction that allows better security analysis, separation of responsibilities, identification

of separate administrative interfaces, and for checking the influence and coverage of legal aspects. It is an architectural framework, because it mainly serves descriptive and explanatory purposes, rather than specification purposes like a reference architecture definition in a standards organization. Furthermore, the reference architecture model is motivated by the lack of support for appropriate security solutions (e.g., suitable monitoring and auditing tools) for critical infrastructure services when being moved into a cloud environment especially when it comes to legal, security and resilience requirements. The architectural model abstracts four different levels: user level, critical infrastructure services, tenant infrastructure, and physical cloud infrastructure. The intention is to observe multidimensional critical infrastructure systems at individual levels from different viewpoints, namely those of multi-provider and multi-tenant, network access, and management, followed by monitoring, policy and resilience orientations. The proposed architectural model is the foundation of this thesis for identifying distinctive abstraction levels, diverse stakeholders and actors per individual level, and identifying vital components within the multi-layered architecture, which is used for building security assurance assessment methodology.

Protecting cloud providers' internal assets against both external and internal adversaries are important just as much as protecting cloud users' information and services against malicious or careless cloud service providers. This thesis highlights that now, more than ever, security and transparency are put to a challenge [IIOW16, OI15]. As much as service providers hesitate to host their services and data in the cloud, the cloud providers are reluctant to disclose any information about their internal infrastructure. This information is only provided under special conditions to the cloud users in form of certificates or audit results, and even then they are very limited when it comes to technical security information. The current solutions for security assessment, certification, auditing or monitoring lack the support for multi-layered environments such as hybrid cloud [Lin16]. Therefore, this work stresses the necessity of building a uniform model that provides automated security analysis by encapsulating in one coherent solution security assessment, certification, auditing and monitoring. The solution should provide continuous security assessment results, in an abstracted manner to prevent sensitive information disclosure and therefore its privacy, but at the same time, offer sufficient information towards cloud users to increase transparency and trustworthiness.

CHAPTER 2

Background

This chapter outlines the most relevant state of the art work with regards to the thesis contributions and details its essential research objectives. First, the security assurance and transparency concepts for the cloud have been evaluated. Next certification schemes that validate corresponding security controls of cloud environments have been addressed. The evaluation of novel monitoring and auditing approaches that are challenged with large scale data sets in distributed system environments was performed. Furthermore, concepts and solutions for assessment of security through security-based metrics models were addressed. Finally, the background section was concluded with the evaluation of the methodologies that define secure service life-cycle from early design to decommissioning.

Security Assurance and Transparency

While some argue that the assurance is the integration of security into the process of designing, building, and testing systems [Lip15], others view assurance as a way to gain justifiable confidence that a service will consistently demonstrate one or more security properties, and operationally behave as expected despite failures and attacks [AADV15]. Nevertheless, assurance is proven to be an indispensable part of computer systems nowadays by increasing both security and transparency [DW15, DPW13, VSR14, BBC+15]. Within this thesis the assurance concepts and methodologies are addressed with regards to cloud computing paradigm. The cloud puts to a challenge the traditional security assessment, compliance, auditing and monitoring approaches by its volatile and dynamic characteristics. In addition, the recent hybrid cloud model [Lin16] poses additional challenges due to its cross-cloud collaborative capabilities. The industrial research conducted by the big four (Ernst & Young [EY16, EY15], PricewaterhouseCoopers [Pri15, Pri16], Deloitte [Del15] and KPMG [Dou16]) puts the problem of transparency and trust in to the spotlight when it comes to IoT, Cloud Computing and Industry 4.0. According to the report [Rig17] from RightScale, hybrid cloud solution is the most preferable approach for enterprises nowadays. This is also aligned with the research community that strives

to address this challenge. According to Ardagna et al. [AADV14], transparency is the foundation of cloud assurance that gives an overview of security status for a cloud infrastructure by delivering security related evidence to a customer. Furthermore, the authors also argue that it is possible to have good security and poor assurance in cloud, due to the transparency towards cloud users. However, often poor assurance goes hand in hand with poor security, which also poses a siginficant limitation when proving that security and privacy properties comply with legislative aspects [AADV15]. In addition, the authors claim that introspection, which is the capability of a cloud provider for examining and observing its internal processes, is not the only concept that matters when considering cloud security. In fact, the concept of outrospection, that is, empowering customers and service providers with the ability to examine and observe cloud providers' internal processes impacting (the security of) their activities/applications/data, is also of a paramount importance. According to their work in [AADV14], an adequate solution to security in the cloud should embrace both introspection by cloud providers and outrospection by cloud customers. Furthermore, authors claim that increased cloud transparency can essentially support the security management problem, supporting both introspection and outrospection. The authors in [SGK+16] performed an extensive assessment of assurance techniques by highlighting required time, man-count, expertise, effectiveness, and costs to perform various assurance techniques. Conclusion of the survey results indicates that the majority of investigated assurance techniques can be completed within 10 days. Furthermore, the majority of the survey respondents also indicated that most of techniques, nineteen of twenty could be performed by at least one or two persons.

Certification Schemes and Security Control Approaches

The fact that security was a major concern, even before distributed computing arrived to the scene, is shown by the Department of Defense that already in 1985 derived one of the first security assessment frameworks for computer systems, the Trusted Computer System Evaluation Criteria (TCSEC) [Lat86]. The book, due to its famous orange covers, was popularly known as "The Orange Book". The TCSEC advocates the security assessment by addressing following fundamental objectives policy, accountability, assurance, documentation and four security divisions (A-D). Each division contains a proposed set of security controls that are validated (e.g. policies, access control models, audit trails, roles, processes, etc.). Unfortunately, most of these controls are focused on confidentiality. Similarly to the TCSEC, the Information Technology Security Evaluation Criteria (ITSEC) [Ran93], founds their assessment model on security controls, but it differentiates assurance and functional levels. The users are able to enumerate the preferred security requirements which the ITSEC formally refers to as Target of Evaluation (ToE). In addition, the ITSEC covers a wider range of security concerns by including integrity and availability requirements as well. One of the most prominent approaches for systematic assessment of complex system environments nowadays, the Common Criteria [Her02], is aligned with the ISO 15408 [fSC08a] and ITSEC [Ran93]. The Common Criteria assessment approach offers a level of confidence that predefined set of security requirements (i.e. functional or assurance requirements) have been met

by the evaluated product formally referred as Target of Evaluation(ToE). Furthermore, ISO/IEC TR 19791 [fSC10] and ISO 18045 [fSC08b] are the extensions to the ISO 15408. The ISO/IEC TR 19791 addresses additional critical aspects of operational systems, and ISO 18045 defines the minimum actions to be performed by an evaluator in order to conduct an ISO/IEC 15408 evaluation.

The compliance of multi-layered cloud environments, that includes both auditing and certification concepts, has been a major point of discussion in a recent few years. European research projects like CUMULUS[1] or CIRRUS[2] delivered very promising certification solutions for cloud environments. The CUMULUS project outlined the first extensive certification model for multi-layered clouds [CDZM13, SDM12]. The approaches introduce only a high level model of the certification concepts that are being used in CUMULUS. Anisetti et al [AAD12] derived some of the first certification concepts that attempts to provide high level of assurance for cloud based environments, as a part of CUMULUS. The authors also proposed a novel certification model for autonomic cloud computing systems based on block-based security certification that they introduced in [AAD14]. They demonstrated their model via service oriented architecture that detects any changes of a service within regards to security, by minimizing test generations and execution activities. Moreover, Anisetti et al continue their work by optimizing performance of the testing activities to an incremental security certification scheme [AAD15a]. Finally, authors in their most recent work [AAD$^+$15b] demonstrate the application of their certification models to an open source cloud solution OpenStack[3]. The certification process was demonstrated by targeting security and performance properties of OpenStack. Also as a part of CUMULUS, Krotsiani et al. [KSM13, KS14] proposed a novel certification methodology for systematic certification of various types of cloud services across multiple cloud layers. Furthermore, authors in their work [KSM13] argue that detailed evidence used for assessment and verification of security can be acquired through continuous monitoring. The authors afterwards demonstrate the efficiency of their certification concepts and present the certification of non-repudiation on cloud storage services [KS14]. Katopodis et al. [KSM14] highlight the necessity of a hybrid certification model for supporting automation of certification and auditing processes in cloud environments by combining testing methodologies and monitoring concepts. The authors argue that such model would significantly increase trustworthiness and security in cloud environments. Bleikertz and Gross proposed a solution for automated analysis of volatile and dynamic virtual infrastructures by crosschecking misconfiguration in the system [BVG14]. The authors in [SS13, WS13, SLGS14, LTSS15] argue the importance of dynamic certification solutions for cloud multi-layered environments by verifying security related concerns across multi-layered infrastructure that are based upon monitoring, security matrices and continuous assessment. Khan and Hamlen in their work [KH13] perform simultaneous

[1]CUMULUS (Certification infrastructure for multi-layer cloud services) `http://www.cumulus-project.eu/`,(last visited January 5, 2017)

[2]Certification, InteRnationalisation and standaRdization in cloUd Security `http://www.cirrus-project.eu/`,(last visited January 5, 2017)

[3]OpenStack - `https://www.openstack.org/`,(last visited January 5, 2017)

security verification process in order to optimize processing costs. Gross et al. [Gro14] propose a very interesting approach that is capable of certifying cloud topology without raveling the underlying infrastructure. Their approach is based upon cryptographic primitives like zero-knowledge proofs [GQ88] that verify security properties.

Monitoring, Auditing and Containerization of Distributed Environments

There is a wide range of monitoring solutions that originate from traditional enterprise infrastructures, cluster, grid and high performance computing, which have been designed or adjusted for cloud environments [WB14]. According to [AADV15] monitoring has become one of the most important aspects of security assurance in cloud as it can also be used to improve the level of transparency towards the users, and therefore the overall cloud security. Nowadays, it is very often a case that service providers and service integrators collaborate in order to provide novel service deployment solutions or build composite service solutions that require automatic service provisioning and composition frameworks. These collaborative cloud environments, the hybrid clouds, offer economies and efficiencies of public cloud models while maintaining the control and security of private cloud models [Lin16]. Hybrid cloud models are opposed by the increased monitoring and compliance challenges due to the rise of system complexity through interoperability. Such frameworks mandate a declarative language as a requirement to describe services, features, and mechanisms for performing provision and composition of appropriate services [TJA10].

Evaluation of the state of the art monitoring solutions for cloud by Aceto et al. [ABdDP13] and Fatema et al. [FEH+14] indicates shortcomings of monitoring tools with regards to scalability, interoperability, multi-tenancy, verifiable measuring and service dependency of both open source and commercial monitoring solutions. The recent work from Ward and Barker [WB14] performs an exhaustive survey of monitoring solutions and derive a taxonomy for examining existing tools and designs for cloud monitoring. The authors conclude their work by examining the socio-technical aspects of monitoring and engineering challenges for implementing monitoring strategies in cloud environments. Hence, despite of the profound features offered by the cloud, monitoring concepts for cloud infrastructure are still faced with many challenges. The current monitoring solutions proposed to address only individual problems such as scalability, large scale data sets, workload, adaptability or availability. This becomes especially challenging when it comes to hybrid multi-layered cloud environments. In contrast to that, Naik et al. [NBVS13] propose a framework for hybrid cloud integration by supporting automation and integration of services across multiple distinct clouds. Rak et al. [RVM+11] in their work propose a solution for monitoring interoperable cloud applications. The authors of [MNP+11], Massonet et al., present a federated cloud monitoring solution based upon RESERVOIR federated cloud architecture [RBL+09] that allows cloud stakeholders to track any operation over their content across the cloud federation. The authors of [vRBV03, BFL+13, SWWM10] propose a hierarchical model that supports segregating workload, whereby the prior

concern of Renesse et al. [vRBV03] and Brinkmann et al. [BFL+13] is to address the scalability challenges of cloud monitoring. Majority of the research community nowadays [NSHS14, dCRdSG+14, APT15, NBVS13, CBR15, vRBV03, SWWM10, RVM+11] favors the agent-based model. The agent-based monitoring model is focused on acquiring information from individual points of interest and storing them on a centralized storage locations. The work of Gonzalez et al. [GMM11] and Casola et al. [CBR15], to the best of our knowledge, are the research minority that addresses the security oriented monitoring for cloud based environments. As such, Gonzalez et al. [GMM11] place the focus on addressing security monitoring of multi-layered environments. Their concept is focused on detecting deviations of initially defined behavior of system elements defined in their policies. In their work [SL16] Singh and Liu also present an approach for cloud monitoring by integrating software frameworks for both batch processing and stream processing frameworks and therefore increase the performance and effectiveness for intensive data trace analysis. Perez-Espinoza et al. [PESSGTL15] successfully demonstrate a distributed architecture model for monitoring private clouds that unifies a set of monitoring tasks. Their approach is unfortunately limited to a single service instance at the time. The authors in [AKG16] evaluated how the hierarchical structure of a cloud affects the monitoring performance with the emphasize on discovering symptoms of malicious behaviors in Cloud environments. Although this research is more focused on malicious behavior within Cloud, it also shows that the performance of current monitoring solutions is highly dependent on the hierarchical structure of cloud services, in particular, if security and resiliency of systems are addressed.

The most recent extensive cloud monitoring solutions surveys [WB14, ABdDP13, FEH+14] address challenges like scalability, interoperability, multi-tenancy, verifiable measuring and service dependency, autonomy, time sensitivity or granularity. Unfortunately they lack to address the container monitoring challenges. The container technologies [PL15, Pah15a] have become increasingly popular because they make significant advancement with regards to automated deployment, scaling, and management applications, consequently imposing a significant momentum especially in cloud [Bre15, BGO+16]. Solutions like Stackdriver [Ciu16] can offer monitoring across distinct cloud platforms (Google Cloud Platform and Amazon Web Services) by giving the insight in to the access to logs, metrics, traces, and other signals from your infrastructure platform(s), virtual machines, containers, middleware, and application tier.

Dana Petcu in its work [Pet14] proposes a visual taxonomy intended to serve the design template for an SLA-based Cloud security monitoring. In [HMC+14] authors propose a SLA, a flexible feature-based model for tenants where the tenants can specify variable requirements. These requirements selectively monitor individual code snippets and prioritize them to determine execution process flow. Their approach is demonstrated on an implemented prototype for multi-tenant service models. Maarouf et al. monitor SLA violations in a Cloud computing environment [AMH15]. The authors demonstrate how a multi-agent system can detect failures, perform self-monitoring operation based on QoS parameters and at the same time operate in a dynamically changing environment.

In their work [RMR+16], Rios et al specify controls and metrics at early design phase and integrate them into a Service Level Agreement. Later a continuous monitoring of those controls and metrics is performed at runtime in the multi-cloud environment. Furthermore, the solution combines deep packet inspection with data mining techniques to harvest and analyze measurements of components and provide holistic assurance. In their work [SME15] authors propose a customizable platform-independent SLA monitoring framework for federated cloud services. Based on the requirements defined in a SLA framework generates monitoring templates which are used to produce reports used for service benchmarking. Casola et al. [CBR15] define specific measurable security-based metrics based on vulnerability scanning and penetration testing and associate them with security Service Level Objectives(SLO) formally specified in Service Level Agreements(SLA).

Another very important aspect of assurance is the ability to observe the behavior of a cloud and evaluate its compliance towards standards, best practices, legislatives and customer driven policies [AADV15]. The auditability of cloud has not only a direct impact on transparency and thus trustworthiness of the cloud users, but also on the privacy with regards to cloud internal sensitive information. Commonly security experts refer to the audit result with regards to standards such as from International Organization for Standardization [fSC05, fSC13, fSC14, fSC15, fSC16] (e.g., ISO 27001, ISO 27017, and ISO 27018), Cloud Security Framework Audit Methods [Sal16], Cloud Security Alliance (CSA) [4] or the American Institute of Certified Public Accountants (AICPA) [5] - Service Organization Controls Type 2 (SOC 2) Report for Service Organizations. Although, the audit reports from such organizations cover a wide range of security controls they are very limited and often not sufficient for answering use case specific challenges. Furthermore, audits also consume vast amount of time and human effort. Recently, cryptographic schemes have become increasingly popular when it comes to performing audits in cloud. In their work Wang et al. [WLL12, WLL15] propose a very efficient approach for privacy-preserving public auditing in cloud environments. The authors use ring signatures to compute the verification information required to perform an audit and protect the integrity of shared data. Data privacy is protected from a third party auditor that can verify the integrity of shared data without retrieving the entire file. In the same context, Cong Wang et al. [WWRL10, WCW+13] propose a secure cloud storage system for supporting privacy-preserving public audits. The authors leverage public key based homomorphic authenticator with random masking to guarantee that third party auditor would not gain any knowledge about the shared data. One of the main approaches when it comes to auditing according to the recent studies [KJM+11, RGYC16, QC13b, MC12, MC13, QC13a, MC14, QC14] is digital forensic. Moreover, the following studies indicate that digital forensic [RGYC16, QC13b] has taken significant momentum especially when it comes to Cloud environments [MC12, MC13, QC13a, MC14, QC14].

[4] Cloud Security Alliance `https://cloudsecurityalliance.org/`, visited 09.02.2017

[5] American Institute of Certified Public Accountants `http://www.aicpa.org/Pages/default.aspx`, visited 09.02.2017

Qualitative Security Evaluation Metrics

Security experts are striving to achieve higher level of transparency and assurance that proper security measures are in place. Quality assurance with regards to security is often performed through penetration [SFM+16] or security requirements [AKF+10] tests. Nevertheless, security metrics play an essential role with regards to security. Unfortunately, it is proven that the security metrics alone are not enough and they must be substantiated [MP16, ASAM12, GJ81]. The essential requirement when assessing ICT assets, which tailors the scope and process of the assessment, is a measurement process and a metric [Jaq07]. Measurement is defined as an act of judging or estimating the qualities of something that can include both physical and non-physical properties of a particular entity.

Authors in [PC10, BB13] highlight the essential challenges for measuring security especially when it comes to cloud [WG14]. In the context of measurement, metric is a standard of measurement derived by a process and therefore can be seen as instrument of measurement, whereby measurement is the comparison of things, usually against standards or best practices [Hay10]. However, building a comprehensive and detailed metric is a challenging task, especially when we take security into consideration. Moreover, it becomes especially challenging to design a metric when we take into consideration volatile and dynamic properties of nowadays ICT environments such as cloud. Nevertheless, Savola et al. in their work [SA09, SSE+15, Sav07] made a major effort for building an efficient security metrics. First, the authors perform an extensive literature research [Sav07] to address security metric taxonomy concepts and challenges for ICT industry. Based upon the literature analysis authors successfully built a security matrices for distributing messaging systems [SA09] supported with a risk-driven assessment [SSE+15]. Luna et al. [GGGS11] introduced in their work first conceptual solutions of security metrics for Cloud based environments. However, the authors derived only a high level systematical overview of the proposed solution that founded the security metric based upon the threat analysis and security requirements. In their work [HSHJ08], Heyman et al. demonstrate their novel security-based pattern approach for performing holistic security analysis derived on aggregation algorithm that iterates through various levels systemically. The approach unfortunately limited the application of their methodology to the development part of the life cycle. Vaarandi et al. [VP14] in their approach had to resolve the large scale data analysis (i.e. big data) because they built their security matrices based upon log file analysis. Caron et al. [CLLT13] were the first ones that developed a security metric, instead of system of scores, with bit vectors as an instrument of their security assessment for verifying individual security requirement at a particular node. One of the important challenges when assessing ICT infrastructures is determining the interdependencies between individual components. To build a hierarchical dependencies, Kotenko et al. [KPSD13] proposed an ontology based metric. This model offered them a possibility to perform the security analysis in a hierarchically organized manner. Sun et al. [SJL+11] introduced an automatic security analysis that illustrated security metrics graphically to reduce the complexity of assessment and at the same time offer a better

overview the system. There are four commonly known measurement scales: nominal, ordinal, interval and ratio [Ste46]. This thesis essentially relates to ordinal and interval scale models, but it also uses threshold model to build the assurance security assessment model.

The process of automated validation of security aspects is entailed with predefined security metrics that were mentioned above. Commonly automated processes are focused on a specific problem (e.g., firewall [AF08], SLA [RGS+16], web service access control [CRV10],etc.). If the cloud computing is observed through its multi-layered abstraction model (infrastructure, platform and service layer) automation process has to cope with a vast amount of challenges (virtual containers, multiple stakeholders, large scale data processing, anonymity, exposing sensitive information, etc.) in each one of those abstraction layers. In addition, each of those layers has be put in to correlation with each other to gain a coherent evaluation result. Therefore, designing and implementing a process that would support automated security validation can be especially challenging when it comes to hybrid environments (IoT [XWP14], Industry 4.0 [Jaz14], hybrid clouds).

Secure Software Life-cycles and Requirements Engineering Approaches

Khajeh et al [KHGS10, KHSBT11] proposed in their work the migration of enterprise IT services to the cloud with regards to context of financial and socio-technical enterprise issues. The decision making process for service migration is conducted by taking into consideration cost modeling and risk assessment. Furthermore, Kaisler and Money, in their work [KM11], evaluate the compatibility of the service migration approach with the cloud computing paradigm by addressing acquisition, implementation, security, usage reporting, valuation and legislative challenges during the process. In their work, Fehling et al. [FLR+13] elaborate and advise best practices for addressing web based service migration challenges with regards to migration patterns.

Requirements engineering plays an important role in secure service development [Nug15, Mea12, SN07], because it identifies crucial security considerations that have to be taken into account during the early stage of development till the deployment and maintenance to ensure the complete service life-cycle. Therefore the requirements engineering has to be taken in to account as a continuous process [GBD+16, QPEM10]. In their work Haley et al. [HLMN08] present a comprehensive framework for security requirement analysis and elicitation based on context analysis. Lipner et al [Lip04, Lip05] designed the most prominent Security Development Life-Cycle (SDL) nowadays, also known as the Microsoft SDL. The Microsoft SDL is used to reduce software maintenance costs and increase reliability by taking software security related bugs into consideration. The work of Busch et al. [BKW14a] is an extension built on the foundations of the Microsoft SDL model, that the authors introduced in [BKW14b] as SecEval. The SecEval framework comprises evaluation process workflow, security context model, data collection model, and data analysis model. Security requirements are defined as constraints on system's functional requirements based upon system or service security goals. Furthermore,

Mellado et al. [MFMP07, MFMP06] leverage the Common Criteria approach for utilizing requirements engineering with respect to security concerns in early development stage, formally referred as Security Requirements Engineering Process (SREP). The SREP is an asset-based and risk-driven model that elicitates security requirements through iterative micro-processes (e.g. identifying, prioritization and categorizing requirements, vulnerabilities and threats, assessing risks, and identifying security objectives). The work of Hesse et al. [HGR+14] outlines an approach that combines heuristics, monitoring and decision documentation to perform semiautomatic security requirements engineering, whereby heuristics monitoring is used to mitigate the manual effort.

Software development is dependent on requirements in order to define and steer how a particular piece of software is being developed. Additional aspects, like security, are also being put into the context through requirements. However, to put the things in a timeline that steers the motion of development, a sequential process or a life-cycle should be established. Nowadays, the motion of development requires the agility to respond to certain changes almost instantly. To fulfill this requirements the widespread concept has been established and used commonly in development projects that is nowadays refered as agile software development [SB02]. One of the manifests of agile software development that supports continuity is continuous integration and continuous deployment model [HF10]. Putting these two aspects in to ac correlation with security (as a process that supports security validation, risk assessment, business criticality, business continuity) and modern hybrid IT environments (IoT [XWP14], Industry 4.0 [Jaz14], hybrid clouds) requires additional evolvement of software life cycle management.

Problem Statement

Cloud Computing [AFG+10] is still an ongoing technological momentum that started as a vision for a novel service delivery model, and essentially became a prominent solution for ubiquitous service and resource utilization. Although founded only on five essential characteristics (on-demand self-service, broad network access, resource pooling, rapid elasticity and measured services), cloud computing rapidly evolved into a global interconnected computing model. One of the main fundamental objectives of the cloud is that the user itself should not be concerned with any activities occurring in background while facilitating cloud services. For example, if a disk fails and users' service was running on that particular disk a cloud provider could rapidly provision a new instance of a service or even keep a clone instance running without user even noticing. Maintenance and provisioning of cloud infrastructure without the users being concerned or aware of any unexpected occurrences on the underlying infrastructures and yet having their service provisioned without disruption is a very attractive and demanding solution offered by the Cloud computing paradigm. Although this model is highly advantageous it is attractive to the users who are not highly concerned with the security and privacy. However, since that is not commonly the case [TJA10], the security (e.g. lack of transparency, data consistency, trust, data loss, control) became the most prominent challenge of the cloud computing paradigm and therefore a huge obstacle towards users hosting their services

in cloud. The unique cloud service utilization eventually steered the cloud evolution towards hybrid model [DMM+12]. Not only does the hybrid cloud model [Lin16, You16] gain more economies and efficiencies over the traditional public cloud computing, but it also brings more control with regards to security. Nevertheless of the deployment model, the cloud computing paradigm adheres the distribution transparency challenges from the distributed systems model [TVS02]. In addition accessing, locating, relocation, replication, and service concurrency still remain unresolved by the hybrid cloud model.

There are types of service providers whose services in case of disruption can confront major subsequent economic loses and affect both other service providers and service consumers on global scale [RPK01]. These service providers are by their very nature considered as critical due to the type of services that they provide (e.g. financial, electric grids, water supply, transportation, health). Therefore they require special attention to assure they are highly resilient by ensuring at any point in time their security, confidentiality, and availability [RPK01, TJA10]. Although that cloud computing could offer notable operational reliability (e.g. high availability, replication and resilience of services and data) and even economic benefits for hosting critical infrastructure services due to the significant lack of transparency and control of deployed services and information, hosting critical infrastructure service in the cloud is not preferred. Therefore, to host such services in the cloud environment, a cloud provider would have to offer high guarantee or assurance of critical infrastructure provider's requirements especially in terms of security. The research community is thoroughly addressing security challenges in the cloud environment [AZ14, SK11, KPR09, ZL12, Pig14, NCAW14, YK13]. It is shown that it is especially challenging when it comes to critical infrastructure requirements because of the high attention towards security [AZ14, YK13].

As already noted above, to convince a cloud user that the cloud infrastructure is operating properly, cloud provider has to offer certain guarantee. Commonly cloud providers offer a legislative agreement of guaranteed services in a form of a Service Level Agreement (SLA) [WBTY11, BYV08] document. Although, SLA empowers legislative strength it does not provide any assurance to the user that certain security, privacy or any other type of mechanisms are enforced or, even worse, not enforced continuously. The SLA unfortunately cannot by any means enforce neither privacy nor security. Nevertheless, the SLA is an important part of the security assurance for defining what measures and mechanisms must be enforced for protecting both cloud users and cloud providers.

This thesis recognizes the *Security Assurance* as a process of integration, continuous evolvement, and validation of security mechanisms. Therefore this thesis argues the necessity to involve the security assurance in all segments of design, implementation, deployment and testing ICT systems and services. Essentially, the security assurance should derive justifiable confidence that an infrastructure and services consistently operate with regards to predefined security requirements, and most importantly increase transparency [Hay10]. Therefore, the security assurance is not only about security as a mechanism, but rather a compound of supporting means like SLA, auditing, certification, monitoring, aggregation and security metrics built into a coherent evaluation model.

Gathering evidence to indicate and prove certain functional, performance or technical aspects have been taken into consideration during the certification process. Therefore, the use of certification approaches has become widely adopted in ICT environments. The certification for cloud environment has been intensively studied in recent few years. The European Union research project CUMULUS and CIRRUS[6] derived a great deal of effort in this domain and established a solid certification schemes for certifying cloud based environments. The certification approaches are challenged by the dynamic, multi-level, volatile and hybrid nature of cloud when building their solutions. Most of the biggest cloud providers nowadays offer hybrid cloud solution[7]. Multi-layered cloud environments can be even potentially owned by multiple stakeholders, and therefore derive a uniform multi-cloud service [PHM+12, SKZ11]. This has been shown to be a promising approach [LBRE14, MZS+14], especially when it comes to offering secure storage services that exploit cryptographic schemes like Shamir Sharing schemes [RST01] to derive secure store solutions like ARCHISTAR [LHS15]. Certification of such environment is a huge challenge, especially because they can be owned by multiple stakeholders that are not keen to mutually share their internal information. This is the challenge that came as the consequence of the hybrid cloud model [DMM+12]. This work highlights that the certification approaches are limited to only a snapshot of the current status of a particular system or service in a certain point of time.

Auditing approaches, due to the fact that they offer essential support in certification process, could be potentially a good alternative for overcoming the continuity problem. The cloud auditing solutions are focused on providing reliable audit trails for supporting legal requirements by having cloud providers privacy in mind [MFB16]. With regards to context of hybrid cloud models, Flittner et al. [MFB16] show the ability to observe the behavior of a cloud and evaluate its compliance to standards, best practices, legislatives and customer driven policies. However, their approach is confronted with information sharing restrictions between cloud providers. Therefore, to conduct an audit of a multi-cloud service we would require a cloud provider independent auditing process and afterwards join the outcome results. This is unfortunately a major overhead for both auditing and certification procedures.

Monitoring solutions have the potential to address the above mentioned challenges of certification and auditing concepts, because they supports assurance by increasing the level of transparency in a cloud with regards to the cloud users, and therefore indirectly security as well. According to Aceto et al. [ABdDP13] and Fatema et al. [FEH+14], monitoring solutions commonly lack support for interoperability, multi-tenancy, verifiable measuring and service dependency in both open source and commercial monitoring solutions. Common focus of the majority of monitoring solutions is on performance and less on security. There are monitoring solutions that support security as well. Two most

[6] Certification, InteRnationalisation and standaRdization in cloUd Security `http://www.cirrus-project.eu/`,(last visited January 5, 2017)

[7] Top 6 Hybrid Cloud Providers `http://www.tomsitpro.com/articles/hybrid-cloud-providers-comparison,2-841.html`,(last visited January 5, 2017)

widespread open source monitoring tools for cloud, Nagios[8] and Ganglia [MLN+12], also support security monitoring (network outages, failed processes, protocol failure, audit and regulatory compliance). Even though, there are monitoring solutions that even address some specific security concerns like network outages, failed processes, protocol failure, they do not counter the overall system security or derive user driven security assessments. Most importantly, when it comes to cloud environment monitoring solutions are challenged with the large scale data analysis and information sharing in a hybrid cloud model. When referring to the Amazon's CTO Werner Vogels statement that Amazon keeps the size of their data centers below 100,000 servers [9] we can easily try to approximate the potential number of components that could be monitored or assessed with regards to security. Given that from each component in such regions as Frankfurt or Ireland we would need to acquire, process and transport information in nearly real time or at least incline to real time assessment, we have to process large scales of data streams in motion. Therefore, we would require a solution that supports processing of large data sets in motion and a solution that can storage acquired information and scale or parallelize its processing.

When it comes to measurement of a particular asset, we require an instrument that will derive a qualitative value of an attribute of that particular asset with regards to a specific concern (e.g., availability, risk, costs). Performing measurements on assets can also be viewed as a certification or compliance process where we have a predefined checklist of points, and each point has to be validated against the particular asset. When observed from this perspective the measurement process can be relatively straight forward. However, when it comes to more complex systems that poses even various types of assets with different owners or stakeholders, such as cloud, mutual interdependencies of each asset have to be taken into consideration as well. Hence, the measurement process easily becomes highly cumbersome. In the approach of this thesis the tools like monitoring tools, scripts, databases, surveys, audit tools or frameworks are designed and implemented to support the security assurance methodology. When we take security as an assessment aspect, a *metric* has to be used as an instrument of measurement process for deriving security condition of a service or a system. In that context, Common Criteria is an example of one of the most widespread information security evaluation frameworks used nowadays, for systematic security assessment based upon security functional and assurance requirements [Her02]. The structural security assessment offered by the Common Criteria derives high degree of flexibility for evaluating an asset with two key components: *Protection Profiles* (PP) and *Evaluation Assurance Levels* (EAL). The PP is used as a set of security requirements for a specific type of asset, such as a firewall. The EAL defines how thoroughly a product is being tested, by scaling it from 1-7, with one being the lowest-level evaluation and seven being the highest-level of evaluation. Unfortunately, common criteria only derives the results of security assessment in a numerical form that indicates

[8] Cloud Monitoring With Nagios `https://www.nagios.com/solutions/cloud-computing./`,(last visited January 5, 2017)

[9] Amazon Cloud infrastructure `http://datacenterfrontier.com/inside-amazon-cloud-computing-infrastructure/`,(last visited January 5, 2017)

completion of a Common Criteria tests. Therefore, we require a more comprehensive solution that supports hierarchical dependency structures and holistic security analysis behavior. One important fact when it comes to assessment process is human involvement as an important and indispensable part of the assessment process. Human involvement is not only involved during the assessment of ICT systems and services, but during the daily operations as well. The problem of human involvement is that it adds uncertainty due to the fact that humans are prone to errors. The shortcomings of the human factor in the assurance assessment process are twofold: human involvement is unfortunately inevitable part of the current state of the art approaches and therefore adds significant time-wise overhead to the whole process [AADV15]; and manual human intervention in complex processes such as security assessment is prone to errors that essentially pose high level of risk to deliver unreliable results or conclusions [Krö08, Bea02, EFL+99]. Therefore this work indicated that building a comprehensive and at the same time flexible security metric is challenging and time consuming task that needs to be additionally supported with structured solution models. To the best of our knowledge, most of the security metrics nowadays require human intervention in their processes which represents a time consuming deficiency that limits the scope of the metric to systems with small complexity.

Another important aspect of security assurance is integration and continuous evolvement of security in the processes of designing, building, and deploying systems and services. Sometimes it is necessary to redesign a service before the deployment. There could be a variety of reasons like costs, design, outdated proprietary technology, because we cannot easily migrate or directly deploy a particular service to a cloud environment. Sometimes applications with performance or licensing concerns are not out of the box solutions prepared to operate immediately in an exclusively virtual world [KGS10]. The processes of design, implementation, build, migration and deployment of systems and services due to their complexity are also prone to failure and therefore require consistency in all segments. The Microsoft SDL [HL06] integrates several phases (training, requirements, design, implementation, verification, release, response) in its cycle, that mainly focuses on development phase of a particular product. Even though, the tool is very flexible and it can also be applied for addressing cloud based applications, it does not directly address cloud based requirements or security requirements engineering in its model. Nevertheless, Microsoft SDL is most broadly used tool for secure development of applications and services. However, for the security assurance it is of major importance to comprehend the security requirements as an iterative evolvement process integrated in such a model as Microsoft SDL. Moreover, the iterative security requirements process (i.e. continuous security requirements engineering) would be of major importance for the security assurance if it would yield the security requirements in such a form where they could easily be reused as measurable security metric attributes. These security properties could easily be used as an input for various monitoring, auditing, certification and assessment policies in the production or deployment phase of a particular asset, bringing all together in a uniform cloud based development life-cycle with regards to security. This has been partially addressed with agile software development process but not when it comes to

cloud, IoT, Industry 4.0 and most important security.

To counter the above mentioned challenges, this thesis aims to provide insights in both cloud architecture and deployment models by extending the current state of the art with regards to assurance assessment methodologies and concepts. One of the main concerns of this thesis is to improve the reliability and reduce the time required to perform the process of assessment by automating the whole procedure. The ultimate goal of this thesis is to develop a comprehensive security assessment framework capable of performing security assessment of the most complex ICT infrastructures, e.g., multi-layered hybrid clouds.

CHAPTER 3

Proposed Solutions

This chapter highlights the core research objectives, outlines methods used for achieving those objectives and details how the evaluation results of this thesis were obtained.

Research Goals

The goal of this thesis is to propose novel security assurance assessment concepts with regards to multi-layered and multi-tenant ICT environments such as cloud. In such a model services can be spanned across multiple abstraction, administrative and even geographical domains. Therefore, such models often implicate additional provisioning, security and transparency challenges. These challenges additionally hinder transparent security assessment of multi-layered and multi-tenant ICT environments. To comprehend these challenges, the main objective of this thesis is set and delivers solutions beyond the current state of the art for security assurance, with a specific focus on following aspects:

- Propose security assurance solutions based on the reference architecture model for hybrid multi-layered cloud environments. In particular, the focus is on multi-layered architectures where each layer can be owned by a different or even multiple stakeholders.
- Designing tools for **continuous** acquisition of security-related information across multi-layered and multi-tenant cloud environments in line with custom or standard driven **policies**. The acquired security-related information will essentially be used to establish a consistent and reliable **audit trail** that supports automated **security assurance assessment**.
- Proposing **structural and holistic security assurance assessment** for performing **qualitative security assessment** of complex multi-layered hybrid clouds assembled from competitive cloud providers. Commonly these cloud providers are

reluctant to share security sensitive information. The security assurance assessment concept has to be aligned with user driven security policies, best practices, security standards, and compliance regulative.

- Implementing novel **security metrics** for supporting the security assurance assessment process.
- Performing feasibility study of a fully **automated security assurance assessment framework** for multi-layered ICT environments via holistic system and service decomposition model.
- Enhancing **service life cycle (design, development, deployment and maintenance)** to support continuous and automated **security integration**, **security requirement engineering** during the whole service life-cycle.

As highlighted in the Background Section 2, the security assessment and analysis processes have been performed in a semi-automatic fashion so far, which largely involved manual human intervention. Moreover, the assessment process commonly involves auditing, certification, monitoring, or compliance validation which undoubtedly offer variety of advantages when used in standard ICT environments. However, when it comes to distributed ICT environments, such as grids or clouds, infrastructures can span across multiple geographical and administrative boundaries, and therefore limiting traditional solutions to perform efficient assessment, aggregation and information acquisition. Furthermore, if the multi-layered aspect is taken into perspective each layer can be owned by a separate competitive stakeholder that is not eager to collaborate with his competitors. Consequently, the tools for security assessment and analysis become highly inefficient. Therefore, the prior focus of this work is finding a solution that will address the problem of assessing, acquiring and sharing security sensitive data between mutually competitive entities in multi-layered cloud environments, and yet offer reliable security assessment method.

Recently, due to its lightweight and performance efficient characteristics containerization technology has become increasingly popular concept in cloud environments [Pah15b]. The result of the emergence of container technologies caused rapid endorsement of composite micro-service models in the cloud because of performance, efficiency to scale, deploy and instantiate services [WLB09, HS15]. Having that in mind, this thesis focused its contributions towards design and development of lightweight information collectors that can easily be deployed as background processes (e.g., daemons) in order to harvest the necessary security related information in a least performance invasive manner. Furthermore, this thesis uses the benefits of lightweight and rapid deployment of virtual containers to integrate the collectors in a complete isolation from a host production environment. In case of a security breach coming from an external adversary the attack would therefore be contained within the domain of a particular container environment. Although, monitoring of cloud systems or services nowadays is well understood, it can easily be misperceived as a straightforward task when taking into consideration resources and capabilities that

can be leveraged from the cloud. Acquiring a specific set of information, especially when it comes to security, is a challenging task that is opposed by many access restrictions depending on the object that is monitored [ABdDP13]. When it comes to internal monitoring, auditing or self-assessments Cloud providers have no restrictions because they own their own infrastructure. However, analyzing own infrastructure and users' services can easily violate users' privacy without users even being aware of it. When it comes to using monitoring as a transparency enhancing tool towards the cloud users, cloud providers are reluctant to disclose any information. Therefore, this work aims to develop a solution that would increase the transparency of cloud provider by performing a security assurance assessment and essentially also increasing security.

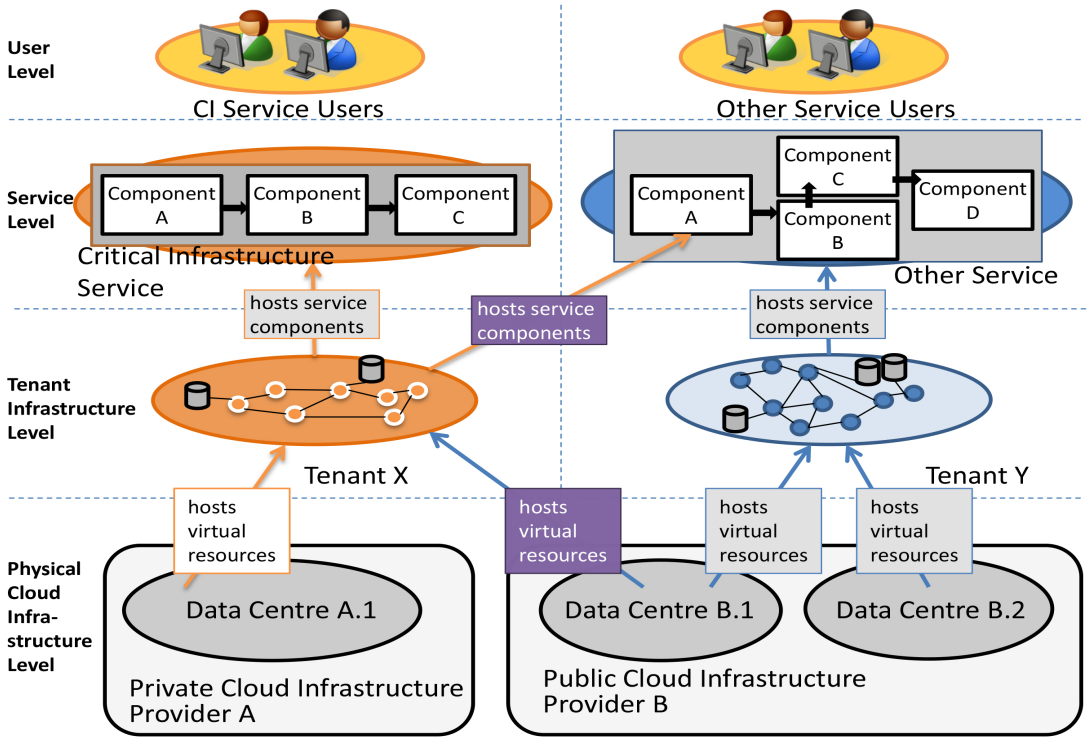

Figure 3.1: Multi-tenant and Multi Provider view of the reference architecture model of SEcure Cloud computing for CRitical infrastructure IT (SECCRIT) research project [BFH+15]

When it comes to structural analysis and assessment of cloud environments, the multi-layered hybrid cloud models were used as a reference model. In particular a structured collaborative cloud solution, i.e. hybrid cloud, that is taking the position as the next generation cloud model was introduced in Section 1, Figure 1.1. This cloud model illustrates how competitive cloud providers can jointly work together to deliver services to their customers. The main advantage of such offering is combining economies and efficiencies of public cloud models while maintaining the control and security of private

cloud models [Lin16]. Dropbox [DMM+12] as a such example of hybrid cloud builds its services on the top of Amazon cloud[1]. More concretely, Dropbox offers its services on a software as a service model towards the users and stores the users' data on Amazon cloud by using Amazon S3[2] storage services on platform as a service level. Having that in mind, the reference architecture in Figure 1.1 that was motivated by the NIST cloud computing architecture, was used to distinguish individual abstraction layers, individual components per layer, and most importantly, their interdependencies. The hybrid cloud notation in the reference architecture gives the ability to differentiate each individual abstraction layer, just as in case of Dropbox, as an independent stakeholder - cloud service provider. The reference architecture in Figure 3.1 illustrates the hybrid cloud service model, where different cloud layers work in conjunction to derive hybrid cloud service. From Figure 3.1 that illustrates conjunction of a private and public cloud provider, we can see that the abstraction and decomposition of hybrid cloud model on its independent components (e.g. virtual machines, storage entities, communication links, etc.), which is essential when it comes to performing structural assessment. Therefore, this thesis analyzes component-wise interdependencies, from the reference model perspective, of both services and infrastructure. The reference architecture remains unchanged, regardless of the cloud solution deployment model and applied use case. The only aspect that changes is the abstraction, administrative or geographical domain for a service. Hence, this is used to define a representative abstraction model that will combine services and infrastructure for the purpose of supporting security assurance assessment. One of the main objectives for security assurance assessment is to flexibly isolate individual components, group of components or components within a particular level of the evaluated system or a service, and apply targeted assessment demands to them. Also, the proposed model should be driven by the compliance regulative, standards and best practices that are used as necessary input in to the security assessment process.

When performing an in depth security analysis, extensive harvesting of security related information is inevitable. In spite of the fact that information sharing among cloud providers in a hybrid cloud would significantly increase the transparency and ease detailed security analysis, cloud providers prohibit any exposure of internal information. As already highlighted, the limitations to perform detailed analysis of services and their underlying infrastructure by the commercial cloud providers is enforced to protect themselves from external adversaries. Therefore, an empirical research of this work is focused towards open source cloud solutions like OpenStack for testing the methodologies and tools proposed by this thesis. Furthermore, the holistic security assurance assessment model is designed by this thesis to offer both on demand and continuous analysis driven by custom based policies.

With regards to the security metric as an instrument of measurement for qualitative security assessment, there are well founded solutions like Common Criteria [Her02] that

[1] Amazon cloud `https://aws.amazon.com/`,(last visited January 5, 2017)

[2] Amazon S3 storage `https://aws.amazon.com/documentation/s3/`,(last visited January 5, 2017)

already offer systematic security assessment solution based upon security functional and assurance requirements. However, the holistic security assessment offered by the Common Criteria offers high degree of flexibility toward the evaluated asset with two key components: Protection Profiles (PP) and Evaluation Assurance Levels (EAL). A Protection Profile is used as a set of security requirements for a specific type of asset, for example firewall, database, disk, or a particular service. The Evaluation Assurance Level (EAL) proposed by Common Criteria defines how thoroughly a product is being tested, by qualitatively ranking a particular entity from one to seven. This work designs a security metric in line with the Common Criteria approach by enhancing it to a wider application domain. The goal is to design a security metric that will be fine-tuned with the security requirements and classes on a more generic level, that Common Criteria refers to as target of evaluation (ToE). In addition, to support the security metric algorithm for hierarchical traversing through the individual elements of a service was designed to inspect each individual security requirement. Moreover, a supportive security policies for individual assets, layers and services in line with standards, best practices and legal regulative were designed and implemented.

Finally, this work highlights the necessity to integrate the security in the service life cycle. Designing and developing cloud services is a challenging task that includes requirements engineering, secure service deployment, maintenance, assurance that proper actions have been taken to support and consider legal aspects. This is especially changeling when the assessment of the global cloud computing model is taken into consideration, which is unfortunately limited due to the current methods and techniques. Hence, this work stresses the necessity for a systematic and comprehensive approach for building such services that starts with the integration of security concerns at the very early stages of design and development, and continuously refine and integrate them through the maintenance phase till the service is decommissioned. This thesis proposes a solution that addresses the mentioned challenges by integrating security requirements engineering and a continuous refinement concept together into a comprehensive security development and deployment life-cycle for cloud services and applications. Furthermore, the focus is also set on iterative refinement of the security-based requirements (i.e. security requirements engineering) during both development and deployment phase.

Methodology

The following methodology was used during both analytical and empirical process of this thesis:

- **Extensive literature review** of the current state of the art security assurance evaluation approaches to identify currently most prominent challenges. The literature review also involved the analysis of different architecture models, policies, regulative, security automation solutions, and cloud models.
- Designing the conceptual model for **security aggregation** and a supporting **hierarchical service abstraction model**.
- Implementation of **conceptual security assurance assessment model** and **security assurance assessment aggregation algorithm**.
- **Prototype implementation** of a proof of concept for security related data extraction techniques and composite security analysis methods.
- **Empirical analysis** of information extraction, transportation, transformation and assessment of expected information increase in security assurance examinations.
- Prototype dissemination via an open source license like the GPL.

The research methodology was established is in line with the work of [Cre02, Kot04], especially with regards to stating research questions and objectives. Data collection and evaluation process have been built on the proof-of-concept model. The data collection process and experiments have been described in the corresponding papers so that they are reproducible in general. Implemented code has been openly published to make the results reproducible as well.

As mentioned in the Section 1, the foundation of this work is based on a reference architecture model, Figure 1.1, that was built for identifying, addressing and developing appropriate solutions (e.g., suitable monitoring and auditing tools) beyond the state of the art for critical infrastructure services when moved or hosted into a cloud environment. This thesis address these challenges for critical infrastructure services with regards to legal, security and resilience requirements. The derived architectural framework introduced in Section 1 was inspired by the NIST Cloud computing reference model. The architecture abstracts four different levels: user level, critical infrastructure services, tenant infrastructure, and physical cloud infrastructure. In addition, to distinguish and address variety of challenges in detailed manner the reference architecture proposes different view models. Hence six independent view models for supporting different aspects have been identified: Multi-Provider and Multi-Tenancy, Network Access, Management-oriented, Monitoring-oriented, Policy-oriented, and Resilience-oriented. The Multi-Provider and Multi-Tenancy, Monitoring-oriented, Policy-oriented views are of particular

interest for this thesis for addressing and comprehending challenges, identifying critical gaps, and proposing potential solutions. These independent views will help us to identify corresponding actors, stakeholders, relevant state of the art solutions and mutual relationships between services on individual layer and underlying infrastructure.

The structural security assessment methodology of Common Criteria framework was evaluated with regards to the architectural framework, Figure 1.1. The initial observation has shown that the structural assessment model and protection profiles design by Common Criteria would seamlessly fit to the structural analysis requirement. The structural approach of Common Criteria is based upon abstraction of individual assets of the evaluated entity as components of evaluation (CoE). For the purpose of this thesis those components are brought together with the corresponding mutual interdependencies and they form a service or a system as a whole that is referred to by Common Criteria as Target of Evaluation (ToE). Furthermore, to perform the security assurance evaluation of each individual component a predefined set of security requirements is proposed. This set of proposed security requirements is defined as a bit-wise security vector. The bit-wise security vector is then used for supporting a novel security assurance aggregation process, that is going to be explained in a while, to evaluate each individual component. Furthermore, a Protection profile from Common Criteria approach is used for identifying best practices of evaluated entities, services or infrastructure. Protection profile identifies most relevant security requirement for a particular asset, e.g. storage device encryption[3]. Furthermore, the assurance assessment approach adheres and enhances the structural security assessment model of Common Criteria by adding elements to the standard structure that allows us to better address the granularity and focus of the proposed assessment approach within a single target of evaluation. In order to visualize the enhanced version of structural assessment model, in line with the service representation and the architectural framework, individual elements and corresponding interdependencies of an evaluated service or a system as general, are illustrated via tree based structured graph [Deo74]. Next, the enhanced structural assessment model is put in to correlation with the general tree based structure to build an algorithm for conducting structural security assessment by traversing through the tree graph. The iterative traversal process that our algorithm enforces essentially performs *security aggregation* through each individual element towards the root of the tree. The security aggregation is conducted by using above mentioned bit-wise security vectors of each evaluated component with the aggregation algorithm and additional polices to derive overall security assurance result. An additional set of policies is designed to support the assurance aggregation algorithm, security properties and assurance level association in deriving final security assurance results in a form of an assurance level.

This work addresses the problem of sharing security sensitive information during the assessment process across multiple competitive stakeholders by referring to a multi-layered and multi-tenant hybrid cloud model [Lin16]. The security requirements in case of a such

[3]protection profile for encrypted storage device `https://www.commoncriteriaportal.org/files/ppfiles/FMV-PP-ESD.pdf`,(last visited January 5, 2017)

hybrid cloud model have to be addressed consistently across different stakeholders, and at the same time consistently aligned with compliance regulation of security standards and best practices without exposing sensitive information of each stakeholder directly. Thus, the bit-wise security vectors are used as the abstraction element for building a consistent and reliable sharing solution of the security sensitive information between stakeholders. Finally, a coherent security metric in line with the Common Criteria assurance levels model was defined. There are four commonly known measurement scales: nominal, ordinal, interval and ratio [Ste46]. The assurance levels defined by Common Criteria derive a result in a form of a level, one to seven, where it is verified that a certain concept has been taken into consideration, e.g., methodically designed, tested and reviewed. On the other hand, this thesis proposes a security metric model based on individual security properties and classes that will be tested consistently across each corresponding component of evaluated system or service. The thesis essentially relates to ordinal and interval scales, but it also uses threshold model. Assurance levels are defined as ordinal scale in terms of values (levels 1-7) that can be compared with each other. However, to classify an assessed entity to an exact level we use interval scale, where a minimum set of conditions have to be fulfilled to be associated with a particular assurance level. The threshold is used on bit basis of the assurance model to define the minimum conditions on bit basis to satisfy the binary one value. This model aims to enforce higher level of granularity towards evaluated services, higher flexibility with regards to defining custom security policies, and enforce more concise and strict security. All of the above proposed assurance concepts build the comprehensive security assurance assessment framework that leads to a higher degree of transparency towards the cloud users.

The empirical evaluation of the security assurance assessment model was performed on a private cloud instance, because of the infrastructure and information access restrictions by the commercial cloud providers. This work is based on the OpenStack open source cloud platform that was chosen for implementing and analyzing the security assurance assessment framework due to its openness, and flexible and modular design. The OpenStack cloud platform allows us to obtain information without any restrictions. To perform the security assurance assessment, security related information must be obtained from each individual component of a cloud with regards to a predefined set of security requirements. Hence, a design for harvesting information across individual layers, components and their interdependencies (e.g., communication links) in a continuous and least invasive manner towards the evaluated service, its components or hosting infrastructure performance was proposed. Furthermore, harvesting information in environments such as cloud where we can have tremendous amount of components consequently leads to a problem of analyzing large scale data. To roughly illustrate the vastness of components, the Amazon cloud provider was used as an example. According to Amazon's CTO Werner Vogels, Amazon tries to keep the size of their data centers under 100,000 servers[4]. Although it is a fairly naive approach, publicly available information of Amazon IP address ranges have been taken into consideration to estimate the potential amount of publicly accessible elements

[4]Amazon Cloud infrastructure `http://datacenterfrontier.com/inside-amazon-cloud-computing-infrastructure/`,(last visited January 5, 2017)

inside a cloud. Furthermore, it is assumed that an individual region is composed of a minimum one data center. Taking the public IP address ranges of an Amazon cloud region and count of particular servers, the potential number of components could be easily approximated (e.g., Amazon cloud region Frankfurt 750,000 and Amazon cloud region Ireland 5,000,000). Given the amount of components in regions as Frankfurt or Ireland we would need to acquire, process and transport information in nearly real time or at least incline to real time assessment, and process large scales of data streams in motion. If we now refer to the scenario for obtaining information from the Amazon cloud region Ireland we would require that in one collecting cycle, i.e., periodic collecting time window, acquire, transmit, collect and process 5,000,000 messages. Having in mind the amount of messages that have to be collected, transmitted, and processed a system, that is able to cope with such large scale data stream in real time is required.

To cope with such a large data stream workload, a distributed messaging and processing system was designed and integrated into the empirical assessment process. The Apache Kafka due to its proven outstanding capabilities to handle large amount of incoming messages [ABD+12, WKS+15]. The Apache Kafka offers rapid scalability and data redundancy with n factor replication that is required by the assurance assessment approach to ensure high availability and enforce resilient data processing. It is demonstrated that the solution proposed by this thesis is capable of handling hundreds of megabytes of reads and writes per second from thousands of clients. Furthermore, the solution offers high scalability where a single cluster can serve as a central data backbone for a large scale organization, elastic and transparent horizontal scaling without downtime, partitioning and spread messages across a cluster of machines for streams larger than the capability of any single machine, and guaranteeing data retention by replication. Apache Kafka clusters incoming messages based upon the meta-information, the message topic identifier, and directs the incoming messages to the corresponding message queues or partitions. With the replication functionality we are able to prevent information loss, by defining the extent of replication. The next step is to process such large amount of incoming messages that are stored in message queues. Hence, due to the to its compatibility with Apache Kafka, the Apache Storm [Ran14, TTS+14] was used as the distributed processing sub-system for the assurance assessment solution. The Apache Storm provides scalability, fault-tolerance of running tasks, and it is easily extensible to provide high performance processing. Furthermore, Apache Storm is a distributed real-time computation system for processing large volumes of high-rate data extremely fast, e.g. over a million records per second per individual node on a cluster of modest size. The information is obtained for each component both continuously (i.e. in predefined time periods) and on demand, by implementing lightweight collectors in Python that will be placed in individual component of the OpenStack infrastructure.

The OpenStack cloud environment was used to emulate the real world cloud environment, but without common restrictions placed by the public cloud providers for harvesting security relevant information. Therefore, it was feasible to perform an extensive analysis across layers and define for each individual element which information should be consid-

ered in the assurance assessment process. The harvested information of an individual component is packaged into a message and transmitted to a cloud infrastructure independent framework where the security assurance assessment is performed. The assurance assessment framework is implemented as a standalone solution that can be hosted both internally inside a particular cloud provider or external as an external third party auditor. This thesis performs empirical evaluation of security assurance assessment framework on a real world scenario that differs in components number and at the same time justifies the realistic number of components and security properties being evaluated. Therefore, three distinct evaluation scenarios are differentiated for the purpose of this work, two cloud-based, and one non cloud-based scenario. The cloud based scenarios are focused on assessing components in cloud data center and therefore Amazon is used as an example of such, whereby two cloud-based scenarios are distinguished as two regions Frankfurt and Ireland that represent two independent data centers. As a third scenario, which is a non-cloud environment, a research center IT infrastructure was chosen to indicate the performance of the approach on a small scale institution (i.e. 1500 employees and only a few dozens of servers). Furthermore, for illustrating realistic number of measurable security properties the following standards, guidelines and best practices were used as the reference: CUMULUS EU FP7 Research Project - 72 Certification Security Properties, ISO/IEC 27001 - Information security management -114 security controls [fSC05], NIST 800-53 - 240 security requirements [NA12], Pay Card Industry Data Security Standards (PCI DSS) - 242 security control requirements [Cou16] OWASP Application Security Verification Standard - 205 Requirements [OWA16]. Furthermore, the performance of the proposed solution is evaluated by evaluating the size and type of messages to see what are the optimal performance requirements, the results are published and listed as research papers in Section 5. Although an instance of an open source cloud test environment was installed for empirical evaluation, due to the modest hardware capabilities, it was not capable of competing with the commercial cloud providers such as Amazon. Hence, for evaluating the performance the assurance framework was deployed into the Amazon Cloud to measure its capabilities in real cloud environments. The empirical evaluation was performed on five Amazon instances that deployed our security assurance assessment implementation with the following configuration: c3.8xlarge instance type[5] , 32 vCPU processors, 244 RAM memory, 2 x 320 GB SSD storage.

Finally, the problem of reliable and consistent design, development, migration and hosting cloud services with regards to security was addressed. The main objective is to integrate security into each phase to enforce automation of security assessment for a particular service or application. The existing solutions like Microsoft SDL [Lip04, Lip05] aim to reduce software maintenance costs and increases reliability. The goal of this thesis is to establish security integration as a reliable life-cycle for design, development and production phase of cloud applications and services. Thus, a comprehensive life-cycle model for supporting reliable service design, development and migration to the cloud environments, cloud development life cycle was proposed. The analysis and then selection

[5]Amazon Cloud instance types `https://aws.amazon.com/ec2/instance-types/`,(last visited January 5, 2017)

of the most relevant security concerns and supportive guidelines, best practices and standards was conducted. The output was used as an input for building a comprehensive survey to seek among industry and academic professionals the most common relevant security topics and concerns. The outcome of the survey outlined the most relevant security concerns in form of security requirements and classes. Then the results of the survey were used to build a taxonomy for a process-based security guideline and to identify relevant security properties for the security assurance assessment process. Afterwards, the development life-cycle was enhanced with the life-cycle of the production part that integrates the security assurance framework as a support for secure maintenance of cloud applications and services, *Secure Cloud Service Development and Deployment Life-cycle*. An important part of the coherent Secure Cloud Service Development and Deployment Life-cycle is the continuous security requirement engineering process that is designed as an iterative evolvement of security requirements, beginning as high level security objectives posed by the users down to the fine grained security properties used for security assurance assessment. These security properties are also designed to be used during the production phase maintenance, auditing, certification, monitoring and security assessment policies. Finally, the flexibility of the assurance assessment framework is demonstrated to interact with other solutions for service and infrastructure assessment or analysis. In particular, it was demonstrated how a solution for inspecting cloud environments by Flittner et.al [MFB16] can be used for deriving reliable audit trails and integrated into our security assurance assessment processes.

CHAPTER 4

Scientific Contributions

The core scientific objectives of this thesis have been divided into two perspectives: security assurance assessment contributions and secure cloud life cycle contributions. The impact of the scientific objectives for this thesis is addressed from the following scientific aspects: cloud provider or cloud user, transparency or security, production or development phase, and practical or theoretical implementation. Hence, this work leverages the analogy of the Cartesian coordinate system to graphically illustrate and argue relationship between scientific aspects addressed by this thesis. Cloud user and Cloud provider are associated to the vertical axis, while transparency and security are associated to the horizontal axis, shown in Figure 4.1. In essence, this thesis argues the impact of each particular scientific contribution (terms marked in blue in Figure 4.1) to the above mentioned aspects. Furthermore, the scientific contributions of this thesis are also illustrated graphically with regards to practical or theoretical application and with regards to which phase (development or deployment). This is again illustrated via the Cartesian coordinate system as shown in Figure 4.2. Firstly, the scientific contributions of this thesis are argued with regards to the correlation of scientific aspects illustrated by the Cartesian coordinate system, as shown in Figure 4.1. The security assurance approach is delivered as a coherent contribution separated in several contribution segments. This is also the reason why the contributions of the security assurance in Figure 4.1 are marked as a rough blue surface shape that covers multitude of domains by four scientific aspects. Furthermore, in Figure 4.1 both, the current state of the art (marked with orange) and the contributions (marked with blue), illustrate how these contributions go beyond the state of the art. When it comes to state of the art, monitoring is placed in the second quadrant because it is mainly focused on the cloud provider for maintaining control over its infrastructure efficiently by identifying certain performance and functional challenges and is more transparency focused solution. Legal regulative, service level agreements, standards and best practices are placed on the vertical axis between cloud provider and cloud user because they impact both aspects, however the impact is mainly

focused towards the transparency since they do not enforce the security mechanism. Microsoft Secure development life-cycle is also placed on the vertical axis, close to the horizontal axis, because it mainly improves the transparency by integrating security into the life-cycle of software development processes. The certification and audit approaches are both placed in the security domain because they validate whether certain security measures have been provided, but still close to the horizontal axis since they support transparency as well. Certification is placed on the borderline between cloud provider and cloud user because it is used to validate infrastructure and services with regards to security. Auditing mainly benefits to the cloud provider, since they are commonly used to internal revision of infrastructures and often not revealed publicly.

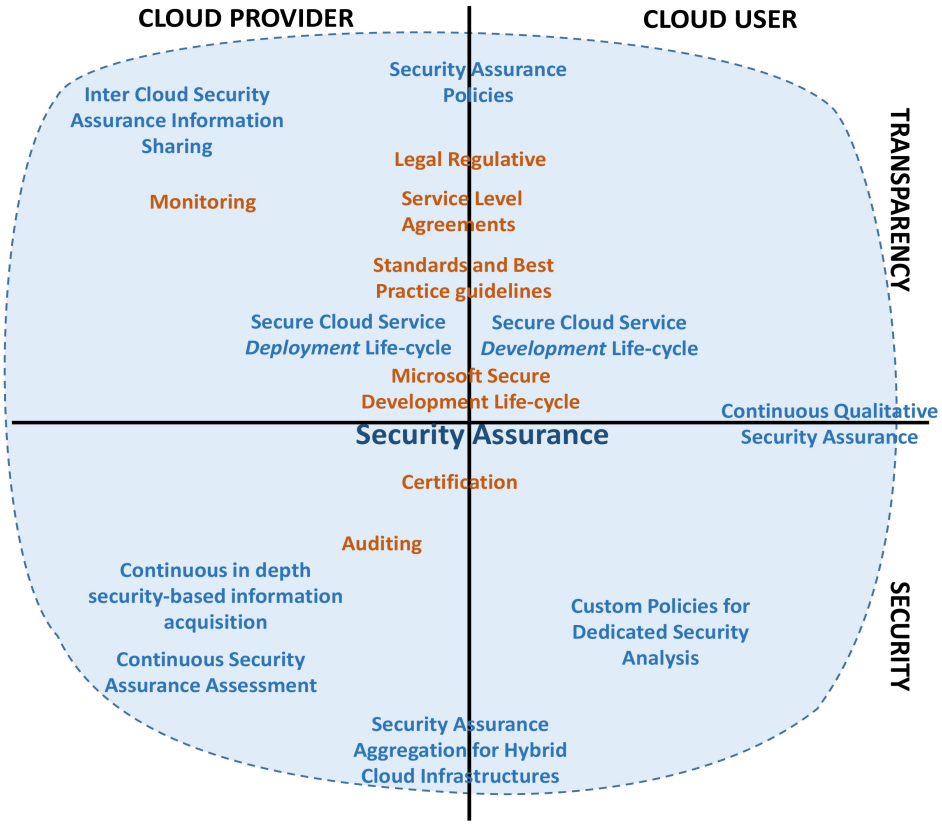

Figure 4.1: Scientific contributions correlation with regards to cloud provider and cloud user against security and transparency. An important note for this graphical illustration is that the axis of the Cartesian coordinate system does not represent exclusiveness between individual quadrants (cloud provider, cloud user, transparency and security).

The secure cloud life cycle [HFL+16, WHM+15] provides two distinct contributions, the secure cloud service deployment life cycle and secure cloud service development life cycle. The secure cloud service deployment life cycle is placed under the cloud provider side because it supports production phase of cloud services. Furthermore, secure cloud

service development life cycle is placed under the cloud user because it enforces more the development part of services before being placed in the cloud environment. Both contributions are placed close to the vertical axis because development life-cycle has to take into consideration the cloud deployment requirements when building cloud services and deployment life cycle has to align the services and systems according to users' requirements. The uniform solution [HFL+16, AHW17], Secure Cloud Service Development and Deployment Life-cycle that encapsulates both development and deployment life-cycles, is designed to enforce transparency by having security deeply integrated into all life-cycle processes.

The security assurance assessment highlights multiple individual contributions that were developed in [HTL+14, HHT+14, AHW17]. The dispersal of those contributions is shown across the quadrants in Figure 4.1. The blue shape marking spread across all four quadrant, Figure 4.1, highlights the contribution areas covered by the security assurance. As presented in [HTL+14, AHW17] this work proposes a systematic approach for continuous acquisition of security related information over multi-layered cloud environments that support the security assessment model. These solutions contribute directly to the cloud provides security by offering the ability to continuously validate security properties across the whole cloud infrastructure, regardless of the geographical and administrative borderlines. To provide a cross cloud aggregated security results a novel solution for component-wise security aggregation was implemented. This particular contribution is placed on the vertical axis between cloud provider and cloud user as it contributes to both by offering continuous qualitative security assessment to users and detailed security analysis to cloud users. The continuous qualitative security assurance is placed on the horizontal axis between the transparency and security for cloud users because it offers both transparency and a unique way to validate predefined security requirements of a user. These security requirements are defined via a custom security policy to provide dedicated security analysis. Therefore, this contribution is placed in the fourth quadrant, since it is focused on cloud users security. Since the security assurance policies can be customized from both sides in line with best practices, regulative and law restrictions, it also contributes towards the transparency of both cloud user and cloud provider, and therefore it is placed on vertical axis. Lastly, the ability to share security sensitive information across clouds during the assurance assessment process directly contributes to the transparency of a cloud provider and at the same time avoids to expose security sensitive information.

Next, the scientific contributions of this thesis are depicted from a technical perspective by putting them into correlation of development and deployment domains against practical and theoretical implementation aspect. Graphical illustration of this correlation is shown in Figure 4.2. The Figure 4.2 highlights the security assurance contribution placed to cover the production side since it is being developed to provide assurance of system and services in hybrid cloud environments. It also covers the development part due to the fact that the framework is developed to support the user requirements and policies that are initially originating from development segment. The security assurance framework is

also supported by theoretical application of policies, security requirements and integrated into a part of Secure cloud service life-cycle. The secure cloud life cycle model that was developed as a part of this work contributes towards both production and development domain, due to the fact that it was designed as a comprehensive framework that supports secure cloud service development and deployment processes. Nevertheless, the production cloud service life-cycle is placed on the borderline between theoretical and practical implementation due to the fact that it integrates and demonstrates effectively how assurance assessment framework can be used in a secure cloud service deployment life cycle [HFL+16, AHW17]. Finally the security assurance policies are placed in the center of the coordinate system as they are involved in all segments of security assurance framework and secure cloud life cycle depicted in Figure 4.1.

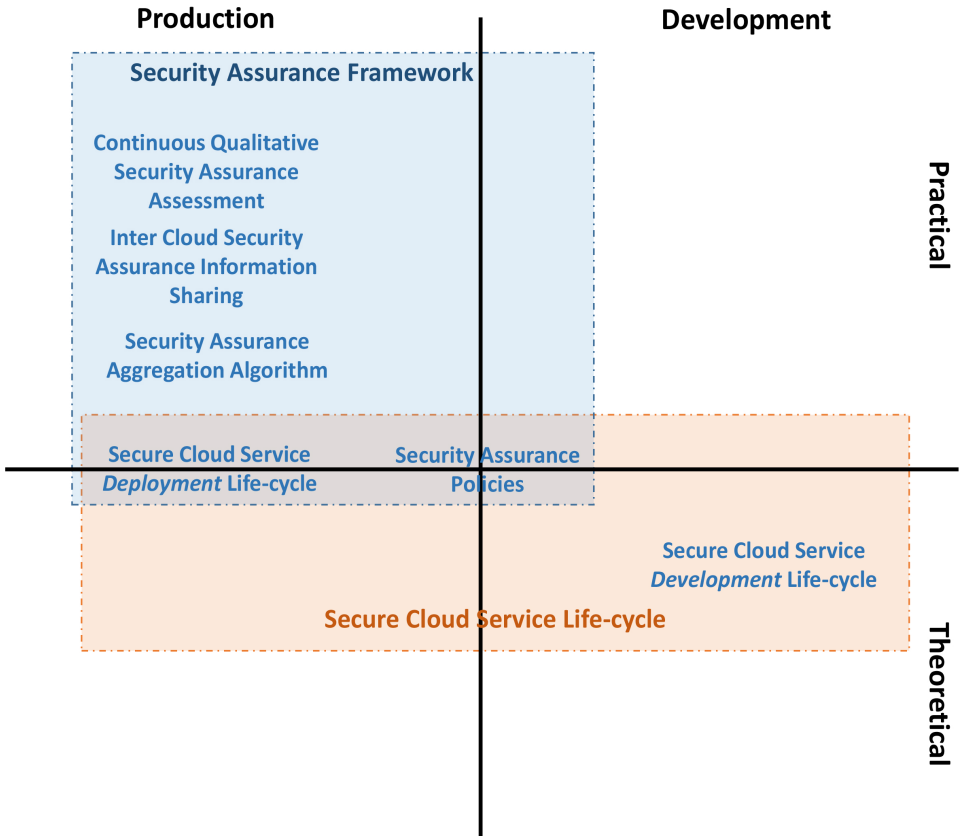

Figure 4.2: Scientific contributions correlation with regards to development and deployment domains against practical and theoretical implementation aspects.

The scientific research and all contributions covered by this thesis have been published as scientific papers. The context of each scientific paper is discussed hereinafter. Each of the paper is classified according to previously identified research fields: security assurance assessment in hybrid cloud environments and secure cloud service life-cycle.

Security Assurance Assessment in Hybrid Cloud Environments

The paper **Towards Continuous Cloud Service Assurance for Critical Infrastructure IT**, published in proceedings of IEEE International Conference on Future Internet of Things and Cloud 2014 in Barcelona [HHT+14], introduces a reference architecture model for hybrid cloud model, identifies shortcomings of the state of the art security assessment approaches, and proposes high level conceptual solution for addressing security assurance. This work argues the necessity for a reference architecture model that will support the abstraction of hybrid clouds with regards to collaboration between multiple layers where each layer can be owned by a separate stakeholder. Furthermore, a reference model architecture was introduced, as a research foundation of this work for addressing multi-layered and multi-tenant cloud environments. The architecture model abstracts service, application, platform, virtual, and physical infrastructure components aligned across four abstraction layers (physical, virtual, application and user). Next, this work highlighted the architecture fine grained distinction of entities along with their mutual interdependencies. This paper identifies major challenges when hosting critical infrastructure service in cloud by performing extensive gap analysis of current state of the art frameworks, best practice guidelines, standards and research projects with regards to the security assessment. The results of the research show that for performing security assurance in cloud, geo-locality and volatility of components are the most prominent challenges. Due to the constant dynamic changes the volatile cloud characteristics inhibit security assessment approaches like certification to provide a plausible and consistent security view. Migration of service that can occur either unexpectedly, due to an infrastructure fault, or planned is not being properly handled by the traditional security or monitoring assessment tools. Conclusively, motivated by the above mentioned challenges, a high level conceptual model based on the Common Criteria approach for building structured security service model for multi-layered cloud environments was proposed. Motivated by the above mentioned challenges and with regards to the general reference architecture, shown in Figure 1.1, this work proposed a service dependency abstraction model and security assurance. The high level conceptual model illustrates how a collaborative and hierarchical security assurance can be achieved, by constructing a model across multi-layered cloud environments based upon service component interdependencies.

The **A Multi-Layer and Multi-Tenant Cloud Assurance Evaluation Methodology**, published in proceedings of IEEE International Conference on Cloud Computing Technology and Science, (CloudCom) 2014 [HTL+14], proposes cross layer qualitative security assurance assessment model with respect to multi-tenant perspective. The solution

that was proposed, is designed by having in mind the following requirements: cross layer assessment, technology independence, information acquisition restrictions, assessment, quantification, aggregation of different information across multi-layered and multi-tenant environments, and most importantly, continuity when performing the security assurance assessment. Therefore, the three major contributions that are the essential elements of the cross layer qualitative security assurance assessment model are hierarchical service abstraction model with the corresponding interdependencies, sequential component-wise security assurance aggregation algorithm, and analytical model for bitwise security assurance aggregation. The generic hierarchical service abstraction model was built by taking into consideration all corresponding interdependencies between service components and aligning them with the reference architecture model in Figure 1.1.

The hierarchical service abstraction model adheres the Common Criteria approach that builds structural security model via evaluation components (CoE) that essentially form a target of evaluation (ToE). The Common Criteria approach was enhanced in this work by introducing a more fine grained structural model for qualitative security assessment that can also be adjusted according to user defined security requirements. The proposed model adds an additional abstraction beside components of evaluation and target of evaluation. This is achieved by addressing a group as an abstraction element that can isolate individual set of components inside of a particular system or a service that are of specific interest to the user. Furthermore, the concept of assurance levels is enhanced, which is in case of Common Criteria designed to validate whether the appropriate methods have been performed or not. The result of the Common Criteria approach is delivered as qualitative assurance levels. The user defined security requirements used to support the security assurance assessment process are engineered in a such a way that they can be used as security validation elements and support fine grained assessment, that was referred to as *Security Properties*. The conjunction of security properties, used as a uniform metric for performing security validation, across all individual components of a particular service is referred as *Security Property Vector*. The whole security assurance assessment model is built around security properties that form custom security metrics and policies used during the security assurance aggregation processes. The paper also introduces a sequential aggregation algorithm that at the very modest level performs aggregation of resulting security property vectors per each component across the hierarchical tree based model that represents a service. The algorithm demonstrates iterative bitwise conjunction of security properties across hierarchical structure in horizontal and vertical fashion. The major objective when building the security aggregation algorithm was to derive abstracted qualitative security assessment without exposing security sensitive information. This work also identified some of the most relevant security concerns with regards to critical infrastructure services that could be potentially used as security properties, like service uptime, data alternation, storage freshness, data redundancy and deletion, or user authentication strength. These security concerns were result of collaborative convergence of security assurance with CUMULUS European research project that is focused on building extensive certification model for multi-layered clouds. The aggregation process supported by a security metric, defines the qualitative value of security assurance by

associating the end result of the aggregation algorithm to a qualitative value of security that was referred to as the *Assurance Level*. Finally, an analytical model for performing theoretical analysis of a service by performing a bitwise conjunction of security property vectors across hierarchical structure model was demonstrated. The model shows how the algorithm iteratively goes through each component and performs bit-wise security assurance conjunction towards the root element of a tree based structure.

The **Security Assurance Assessment Methodology for Multi-layered Clouds** paper was successfully accepted at the Elsevier Computer and Security Journal [AHW17]. This paper is a follow up work of the [HTL+14] where a theoretical concept for performing security assurance assessment was implemented. The theoretical concepts proposed in the previous work was enhanced by implementing custom and policy driven security assessment approach. This approach integrates security engineering and supports a holistic service and infrastructure abstraction for geographically and administratively distinct environments that can be owned by different stakeholders. Furthermore, the focus of this work is to preserve privacy of the assessed systems and at the same time offer lightweight information acquisition with minimal impact on the service performance. The ultimate goal of this solution is to process and share large scale security sensitive data across multiple clouds. This is supported with the empirical evaluation of assurance assessment solution.

First, a simplified real-world service use case scenario (video surveillance service) was introduced. This work performed a formal abstraction and decomposition of the above mentioned use case on independent components. The use case service is then illustrated as an abstracted service model in a hierarchical tree-based form with a root component as the top of service. The security assurance aggregation algorithm felicitates supportive assurance entities (security properties, security property vector, evaluation set) in its assessment and aggregation process steps. These assurance entities are then used in conjunction with the assurance assessment algorithm to derive security assurance result for the particular service. In addition, a supporting component-wise aggregation policies are defined on a group and level basis. The level aggregation policies are focused toward components placed at a certain level of interest, regardless of whether they are located in a single administrative domain or not. The group policies address both the components that are placed on different hierarchical levels, and components that are on the same horizontal level (e.g., physical servers). The flexibility to identify a specific group of components within a single service allows us to identify predefined set of requirements that apply to this specific group of interest (e.g., physical servers). This work introduced a specific policy to derive the end result in a form of numerical level, the assurance level association policy. The assurance association policy is essentially a security metric that associates the aggregated result, which is a binary security property vector, to the corresponding security assurance level. The flexibility of associating different levels together with predefined corresponding set of security properties are used for service validation facilitates protection profiles. The protection profiles, unlike Common Criteria, can be therefore facilitated from the perspective of a whole service (e.g., video surveillance

service) or domain (e.g., healthcare data protection) and not only on component basis. The ability to abstract the end results of the security assurance assessment as a bit-vector, i.e. assurance level, prevents the exposure of internal security sensitive information. Therefore, this information can be shared across multiple, even competitive, cloud providers and calculate a coherent result. This was demonstrated by applying the assurance assessment model on analytical model first.

Furthermore, a novel implementation of security assurance assessment proof of concept for open source cloud environments was introduced. The implementation of security assurance assessment framework was done by using the OpenStack open source cloud platform as the foundation of this research due to its composite and modular characteristics. This allows the flexibility when investigating and harvesting information across the complete cloud infrastructure without restrictions. In spite of the fact that the aim of this work is focused only on cloud based environments, it is also applicable to a wider set of ICT environments (e.g. grids, clusters, virtual environments) with layered and composite characteristics. The solution can easily be deployed and provisioned as an external and internal assessment tool. The assurance assessment framework by design offers high availability and resilience to failure through its system segments, in particular messaging and processing, with regards to processing acquired information. The solution integrates five essential sub-systems (information acquisition, messaging, processing, storage and presentation) as a part of the security assurance framework. Moreover, design supports users' demands at optimal costs by performing assessment process in the following three operational modes: *continuous* by performing the security assurance assessment in discrete predefined periodic time intervals, which is referred to as collecting cycles; *on demand* by performing the security assurance assessment at certain points based upon users' needs (i.e. during audits); and *event based* by performing the security assurance assessment at the point when a change in the system occurs. Information acquisition system is implemented as Python modules that extract the raw data from a CoE and construct JSON messages that are handed over to the messaging system. The messaging system is based on Apache Kafka distributed messaging system [ABD+12, WKS+15] to ensure reliable message handling, due to its ability to offer rapid scalability across multiple brokers and clusters, and data redundancy. Due to the high compatibility with Apache Kafka distributed messaging sub-system, Apache Storm [Ran14, TTS+14] was chosen as the distributed processing system. The design is fully capable of supporting auditing, certification, compliance validations and monitoring with regards to security in a highly efficient manner. This is shown by the performance capabilities of the proposed concept that shows significant improvement with regards to assurance techniques identified by Such et al. [SGK+16] that sometimes require even several days with manual human intervention to finish.

Secure Cloud Service Development and Deployment Life-cycle

The paper **Impact of Critical Infrastructure Requirements on Service Migration Guidelines to the Cloud** [WHM+15] that was published in proceedings of IEEE International Conference on Future Internet of Things and Cloud 2015 addresses the relevance of individual security aspects for different industry and academic stakeholders when migrating services to a cloud environment, with emphasis on critical infrastructure providers. This work also analyzes the coverage of required concepts and approaches for increasing trust and security in a cloud environments by industry standards, best practices and guidelines. In addition, the paper proposes a taxonomy for secure service migration toward cloud environments as coherent life-cycle. This work also performs an extensive survey among industry and academic professionals to investigate the security related concerns with respect to critical infrastructure security requirements, that can afterwards be used to support security assurance assessment. The focus on the critical infrastructure service providers was attractive due to the criticality of the services. The importance of cloud requirements was investigated by this work with regards to the NIST cloud computing characteristics. In particular, the focus was put on transparency aspects of the cloud by addressing geolocation, resource pooling, on demand, services, rapid service provisioning, measured service and ubiquitous network access. In addition, typical information security requirements such as availability, integrity, confidentiality, auditing according to their relevance for critical infrastructure services with regards to facilitating cloud computing services were addressed. The outcome of the survey showed high concerns towards the geolocation of services and information, and increased tendency towards the confidentiality and integrity of information and services. Furthermore, the survey outlined that transparency concerns also relate to risk assessment, incident response, SLA management, use of international standards and service life cycle. The analysis of industry standards, best practices and guidelines for designing and implementing secure services indicates deficiencies with regards to security controls that was identified as relevant through the survey analysis and state of the art research. The outcome of the extensive analysis and survey results was used to support information security taxonomy that was built for secure service migration towards the cloud. The taxonomy proposes a sequential process for migrating services securely to the cloud environments through five essential stages: analysis, design, implementation verification and deployment. Each individual step of the life-cycle is aligned with a set of requirements originating from industry driven guidelines, standard, best practices, and legal recommendations, when migrating services securely towards the cloud.

The paper **Towards a Unified Secure Cloud Service Development and Deployment Life-cycle** was published in proceedings of IEEE International Conference Availability, Reliability and Security (ARES) 2016 [HFL+16] and is a follow up work of [WHM+15]. In this work the standard process-based migration guideline was enhanced to a comprehensive secure cloud service development and deployment life-cycle. First, a security requirements engineering model was proposed. This model starts with defining

high-level user or use case driven objectives that are iteratively refined towards fine grained security requirements and properties. The process involves context analysis aligned with industry driven guidelines, best practices, international standards and legal requirements that can essentially support various policies for monitoring, assessment, auditing or even design and implementation. A two-phase secure cloud service life-cycle process that integrates requirements engineering and iterative refinement with respect to security was proposed by this work. The first phase is called *Development phase* and covers the sequential set of steps where a service is being designed and developed. Secondly, the *Production phase* is where a deployed service is validated against security requirements that have been defined in development phase. The approach narrows down the focus of the secure development life-cycle process by taking the following objectives into consideration: integration, engineering and continuous refinement of security requirements in design, development, testing, deployment and maintenance software for cloud based architectures, migration of legacy software system to the cloud, iterative security requirements engineering during both development and production life-cycle phases. Via an application scenario, this work demonstrates the integration of supportive tools into the secure cloud service development and deployment life-cycle with emphasis on the assurance assessment framework. In order to ensure that security requirements have not only been properly integrated into a service during development but that they are also properly ensured during the deployment, integrated security assurance assessment framework continuously monitors key security aspects. The assurance assessment framework uses as input the user or use case defined requirements, in form of security properties, and verifies them across the deployed environment continuously.

Real-World Appliance

Although the research of this thesis is mainly focused on cloud environments it can be applied in a much broader domain in context of hybrid infrastructures like IoT [XWP14] or Industry 4.0 [Jaz14]. To bring the methodology, security metrics, algorithms and concept developed by this thesis into the context of the real-world challenges, an illustrative use case example is used to show the advancement that this research brings.

A pharmaceutical drug track and trace model [KSCB03, LCTCY05] is used to demonstrate how the work of this thesis can cope with the real world problems. The use case will bring together smart solution processes like Manufacturing, Warehousing, Transportation, Healthcare and Cities into an advanced coherent system for drug production, distribution and consumption. The drug track and trace model is going to be steered and supervised from a central cloud solution hosted in a public cloud environment. To illustrate the application of the work presented by this thesis by a real world scenarios an Industry 4.0 use case model will be elaborated. Industry 4.0 foresees that the digital transformation has taken a large momentum when it comes to complex supply chain model that facilitates an interconnected model for manufacturing, warehousing, transportation, distribution, delivery and purchasing processes. Each of the mentioned processes in the context of Industry 4.0 requires at the same time strong autonomy and collaboration between

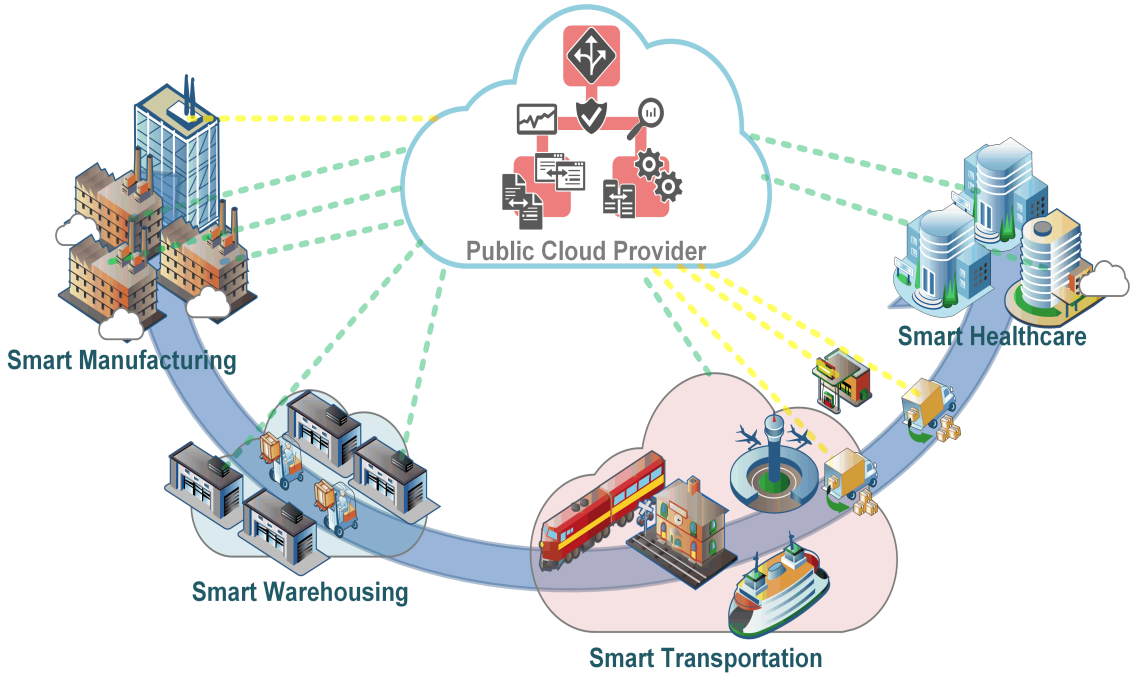

Figure 4.3: Smart Use case model in context of Industry 4.0

processes. The use case embraces different segments of smart solution models like Smart Manufacturing [LIS16], Smart Warehousing [Din13], Smart Transportation [WRC+16], and Smart Healthcare [CdDM+15], to illustrate the complexity of systems and services nowadays. Drug track and trace model is used to efficiently prevent drug counterfeiting, ensures the control of illegal procurement and eases the drug revocation in case of defect products. However, to achieve this, sensitive healthcare information is shared through the whole drug life-cycle, (i.e., across several systems or stakeholders).

The Smart Manufacturing model within context of drug production [LMB09] implements a comprehensive production line process which mitigates human errors, prevents drug counterfeiting and increases efficiency by completely excluding human as a part of the production process. To achieve this, smart manufacturing model facilitates semi- and full-automation processes in production. The Smart Warehousing model[Din13] implements functionalities of inventory management, logistics, packaging, and preparation for distribution and transportation. The transportation is the next smart segment of the use case chain that has to be properly addressed so that the products reach their end destination in time. Lastly, a smart Healthcare smart model is facilitated. The process for tracking and tracing drug products through the whole life-cycle is established via a centralized cloud based solution.

Simultaneously to manufacturing, warehousing, transportation, distribution, delivery and purchasing processes, the whole product life-cycle of the product must be consistently

monitored and supervised so that each individual segment satisfies best practices, ensures security and privacy, and is compliant with strict standards. Common practice in industry is to perform regular audits and certifications to ensure security assurance aspects against a particular use case. The research of Such et al.[SGK+16] evaluates the required effort for performing security assurance aspects (penetration tests, architecture validation, vulnerability scan, code analysis, fuzzing, threat assessment, auditing, compliance validation, etc.). Their research results indicate that a significant effort with regards to time and human intervention is required to contemplate and perform validation of security assurance aspects. When it comes to complex environments such as cloud or the above mentioned smart model use case, the required effort cloud be easily measured in months. If only monitoring is taken into consideration, it is put up to a challenge when it comes to such heterogeneous environments. The use case involves different infrastructure segments that cover manufacturing robots, servers, virtual machines, vehicles, containerization, embedded devices or sensors, and therefore it is very unlikely to be monitored with a single solution.

This becomes especially challenging when taking security into consideration that requires a certain degree of comprehensiveness when assessing. The common way of addressing security requirements in such cases is to validate compliance to security standards like ISO27001[fSC05], ISO27002[fSC13], ISO27017[fSC15], ISO27018[fSC14], ISO19086[fSC16], Cloud Security Alliance [1], or Service Organization Controls (SOC) 1-3 [2]. Additional problem with regards to security is the challenge to automate security during the process of design development, deployment and maintenance. The focus of this thesis is to minimize the effort and increase the efficiency for security assessment, especially with regards to continuity and real time evaluation. Therefore with regards to the above mentioned use case, the following challenges are highlighted:

1. Sharing sensitive information among different stakeholders to support holistic security evaluation.

2. Impact on performance in sensitive and resource limited environments like containers and embedded systems.

3. Performing nearly real time security assessment across multi-stakeholder environments.

4. Hierarchy-based security assessment decision model.

5. Process automation with regards to development, deployment and security validation across the whole life-cycle of a service or a system.

[1] Cloud Security Alliance `https://cloudsecurityalliance.org/`, visited 09.02.2017

[2] Service Organization Controls (SOC) Reports for Service Organizations `https://www.aicpa.org/InterestAreas/FRC/AssuranceAdvisoryServices/Pages/SORHome.aspx`, visited 09.03.2017

Each of the above highlighted challenges are going to be addressed below by the individual contributions covered by this thesis.

Sharing sensitive information among different stakeholders to support holistic security evaluation. In the healthcare use case, ensitiven information are bing hosted across several ICT environments through the whole value chain. Therefore, each individual segment of the value chain has to ensure that proper security measures have been met. Therefore, the easiest approach is to analyze each individual segment separately and then aggregate the results. Since this requires sharing infrastructure internal sensitive information between stakeholders, which can also be competitive, this poses a great threat to security.

As already mentioned above, the common way of addressing security requirements in such cases is to validate compliance to security standards. With regards to the above mentioned use case this approach entails significant effort and consumes a lot of time, because each segment has to be analyzed separately and then a consolidated manual assessment has to be conducted. The holistic security model proposed by this thesis offers a way to address security of each individual segment and share it in a transparent and non-invasive manner with regards to privacy (i.e. exposing internal sensitive data of a particular segment). This is conducted by introducing a bitwise vectors where each bit of the vectors represents a security requirement (e.g., encryption, strong authentication, strong password, etc.). This vector, by using the proposed security aggregation algorithm, namely security assurance algorithm, first performs security assessment of an internal segment separately. Therefore, preventing any unnecessary exposure of internal sensitive data. Afterwards, the results of each segment are taken into the final assurance assessment and the results are consolidated to gain the final resulting vector. The resulting vector is then used to determine Security Assurance level of the whole use case.

Impact on performance in sensitive and resource limited environments like containers and embedded systems. Analyzing for example smart production or manufacturing environments can be challenging because very often there are systems like containers, sensors, robots that possess limited amount of computational resources. Therefore, such environments require special care to avoid any impact on the performance.

To mitigate the impact on the performance of performance sensitive environments the thesis addresses the problem of information acquisition process by analyzing different message structures. The message structure was optimized to carry the minimally required information set that can be used to make bitwise security requirement compliance decision.

Performing nearly real time security assessment across multi-stakeholder environments. As already mentioned above, analyzing and assessing not just security of such a complex system model as shown in the Figure 4.3 requires a great deal of effort and resources. Traditional IT systems unfortunately do not offer such a degree of details when it comes security of distributed systems.

Therefore, by combining the first two points with a distributed messaging and processing solutions this thesis was able to arbitrarily scale on demand the workload. Therefore, for demonstrating capability to address security assessment of several million IT components in manners of seconds, depending on the number of evaluated security properties, the security assurance assessment model was based on scalable solutions.

Hierarchy-based security assessment decision model. To propagate the security assessment across different multi-stakeholder and multi-layered environments, the thesis developed a holistic security abstraction model that can segregate security assessment results. The results can be segregated based on a predefined group of interests and finally perform consolidated security assessment. This holistic abstraction model allows the particular service to be distributed or hosted by several independent segments just as in the smart use case model above. As already mentioned, the results can be consolidated across several administratively separate environments without violating any sensitive information exposure.

Process automation with regards to development, deployment and security validation across the whole life-cycle of a service or a system. The momentum of the technology evolution made a major impact on software development process. Old traditional processes like waterfall, iterative, spiral or incremental process for developing cumbersome software has been replaced with a more dynamic agile development process focused on microservices. Moreover, the change of development process has also brought continuous integration and continuous development models that are widely established in developing both cloud and enterprise services. Most recently, the automation of development and deployment has taken a large momentum to avoid any manual human intervention and increase the performance of processes. However, harmonizing these processes into a single dynamic life-cycle is still a challenging task. Therefore this thesis implements a comprehensive security based design, development and deployment process supported with requirements engineering to address them and simultaneously automates security in to the whole software life-cycle.

CHAPTER 5

Conclusions

This chapter concludes this research by outlining contributions, limitations and potential future work. The contributions and their implications towards the advancement of Cloud management, monitoring and security assessment, as well as the application of novel security assurance metrics in the context of hybrid cloud environments are summarized in this thesis. As already mentioned, this research is founded on the reference architecture models introduced in Section 1 in Figure 1.1 for addressing multi-layered and multi-tenant cloud environments. The architecture model abstracts service, application, platform, virtual, and physical infrastructure components across four abstraction layers (physical, virtual, application and user). Therefore, the architecture enables a fine grained distinction of entities in a component wise manner together with their mutual interdependencies that support the assurance assessment framework.

Summary

Within the scope of this thesis a generic cloud security assurance assessment model is designed and implemented. This model offers security based monitoring for both cloud users and cloud providers, capable of being deployed across multi-layered and multi-tenant cloud services in a modular fashion. The security assurance assessment model uses a novel security metric for component wise security assessment which includes mutual interdependencies across different abstraction levels and components. At the same time, this approach ensures protection of internal security sensitive information from exposure. Furthermore, a scalable security oriented monitoring and assessment solution capable of withstanding large scale data streams is implemented. The security information monitoring and assessment is utilized by acquiring information sets across each individual component in a cloud environment regardless of the layer or stakeholder.

Security Assurance Assessment for Hybrid Cloud Models. For the purpose of the holistic security assurance assessment a **generic tree based structural model** was introduced to depict service components and its interdependencies. The model is designed to illustrate service components and mutual interdependencies in a hierarchical fashion so that the root element represents the top level service commonly offered towards the cloud user. The tree based model was used when designing the **security assurance assessment algorithm**. The algorithm is facilitated in the implementation to provide continuous qualitative security assurance by offering both transparency and a unique way to validate predefined security requirements of a user. The security requirements are built in a custom security policy to support extensive **security assurance aggregation** concept. This work also demonstrates that detailed security assessment of complex ICT infrastructures such as cloud, require time, personnel, expertise, effectiveness and usually come at high costs. Furthermore, the current solutions for monitoring and assessing security fail to address scalability. Therefore, this research introduces an integrated monitoring and assessment solution that implements separate entities for analyzing, storing, and queuing to support scalability. The solution that was proposed by this thesis is designed based on the following requirements: **cross layer assessment, technology independence, information acquisition restrictions, assessment, quantification, aggregation of different information across multi-layered and multi-tenant environments,** and most importantly, **continuity when performing security assurance assessment**.

Secure Cloud Service Deployment and Development Life-cycle. Engineering services to adopt cloud requirements is a cumbersome and time consuming process, especially if security is taken into consideration. Commonly the process of design and implementation of a service starts with high level objectives that should be argued with security concerns. Both, concerns and objectives have to be integrated into a service, as well as the cloud requirements. There are also many applications that were developed before the cloud paradigm and have to adjusted before deploying in a cloud environment. Hence, a life-cycle model was built firs to support reliable service development and migration to the cloud environments, cloud development life cycle. Selection and analysis of the most relevant security concerns and supportive guidelines, best practices and standards was conducted as a part of this work. The development life-cycle was enhanced by integrating the production phase. The security assurance framework was embedded as a supporting toolset for secure maintenance of cloud applications and services. Furthermore, an important part of the coherent Secure Cloud Service Development and Deployment Life-cycle is the continuous integration of security requirement engineering process that is focused on iterative evolvement, beginning as high level security objectives set by the users down to the fine grained security properties used for security assurance assessment. These security requirements are used through both life cycle phases as an input for various monitoring, auditing, certification and assessment policies in the production or deployment phase of a particular asset, bringing it all together in a uniform cloud based development life-cycle with regards to security.

Limitations

In this section, the limitations of the scientific contributions are addressed and presented. These issues are important for proper understanding of the core objectives of this thesis and highlighting observations that were out of the scope of the research.

- The security assurance assessment framework represents a first version of the proof of concept model for continuous security assessment in complex ICT environments. The current implementation is limited to the scaling of a particular segment where assurance level is calculated for complex tree models. The optimization of processing assurance level for complex tree structures was not addressed due to the fact that it was not the main objective that supports security assurance concepts.
- The proposed set of security properties represents just an initial set of properties used for supporting the security assurance assessment model and building a proof of concept. The research was focused only on technical aspects of security for building qualitative metrics.
- The security collectors are for the scope of this thesis limited to the automatic management and deployment. Therefore, it is necessary at the installation point of each component to integrate the collectors manually as well. Furthermore, activation of individual collectors is currently implemented via configuration file which limits them for large scale automated deployment.
- The implementation introduced in this thesis establishes a modest audit trail by using a SQL database to dump incoming messages without taking into consideration the performance effort. Storage module performance can be additionally improved due to the ability to flexibly scale the process by proceeding the messages either to Kafka messaging system or to a more performant storage system (e.g., No-SQL, disk appending, etc.).
- The interoperability of secure cloud service deployment and development life-cycle has only been addressed with regards to security assurance assessment framework and cloud inspector. In addition to the limited amount of security properties being addressed no actual deployment of policies has been derived by the life-cycle.

Future Work

As highlighted in the previous section, some of the approaches proposed by this thesis have some limitations because some issues are still being out of the scope for this thesis. Therefore, they were not covered here. Nevertheless, they still represent very important shortcomings and therefore they are left to be address as a part of the future work. Here are some research topics of this thesis that are going to be address as a part of the future work:

- The assurance assessment framework covers the basic set of security properties for supporting proof of concept model. An extensive elicitation of security properties was performed with regards to security standards, best practices, and hosting platforms. Furthermore, these security properties will be used to support cloud service protection profiles. Cloud service Protection profiles are going to be designed for different domains in line with legal compliance, best practices, standards and policies. Additionally, testing and expanding the approach on different virtualization (e.g., Xen and VMware) and cloud environments (e.g., CloudStack, OpenNebula, Amazon, etc.) would provide more comprehensive security metrics, protection profiles, security policies.

- As discussed in the previous section this thesis is only focused on the manual installation of the collectors during the component's instantiation time. Facilitating a centralized management and deployment solution for collectors would significantly reduce both maintenance and administration cost for large scale cloud deployments.

- Maintaining a consistent and reliable audit trail is to highest importance of the security assurance assessment framework. However, the performance and scalability of that particular segment, due to the widespread and well established community was not therefore the highest priority of this thesis. Also, one of the future goals is to enhance the solution by investigating the potential storage solutions like No-SQL databases, appending the data on the high speed disk drives, adding additional Apache Kafka messaging queues, etc.

- The continuity aspect of security assurance assessment module gives us the security status and shows any changes with regards to predefined security requirements of the system or a service almost instantaneously. These changes or deviations are going to be used to enhance the security assurance assessment framework as an anomaly detection tool. Additionally, the utilization of machine learning concepts could be used to identify those deviations and act correspondingly.

- Finally, the multi-layered and multi-tenant perspective of the model that we applied in hybrid cloud model could potentially be enhanced to support next generation ICT systems, like IoT and Industry 4.0, due to their interoperability characteristics. Therefore, the appliance of the approach towards these systems is going to be investigated as a part of future work.

CHAPTER 6

Research Contribution Overview

Publication List

Publication list in a chronological order.

1. Hudic, A., Hecht, T., Tauber, M., Mauthe, A., & Elvira, S. C. (2014, August). Towards continuous cloud service assurance for critical infrastructure IT. In Future Internet of Things and Cloud (FiCloud), 2014 International Conference on (pp. 175-182). IEEE.

2. Hudic, A., Tauber, M., Lorunser, T., Krotsiani, M., Spanoudakis, G., Mauthe, A., & Weippl, E. R. (2014, December). A multi-layer and multitenant cloud assurance evaluation methodology. In Cloud Computing Technology and Science (CloudCom), 2014 IEEE 6th International Conference on (pp. 386-393). IEEE.

3. Wagner, C., Hudic, A., Maksuti, S., Tauber, M., & Pallas, F. (2015, August). Impact of Critical Infrastructure Requirements on Service Migration Guidelines to the Cloud. In Future Internet of Things and Cloud (FiCloud), 2015 3rd International Conference on (pp. 1-8). IEEE.

4. Hudic, A., Matthias F., Lorunser, T., Philipp R. and Roland B. (2016, August). Towards a Unified Secure Cloud Service Development and Deployment Life-cycle. In Availability, Reliability and Security (ARES), 2015 10th International Conference on (pp. 501-510). IEEE.

5. Hudic, A., Smith P. and Weippl E. Security Assessment Methodology for Multi-layered Clouds Elsevier Computers & Security Journal

Released Source Code

Assurance Collectors: `https://github.com/Austrian-Institute-of-Technology/seccrit_collector`

The Assurance Assessment Framework code is not fully published due to the rights of private code property of Austrian Institute of Technology.

(Co-)Instructed Master Theses

Martin Kirchner master thesis at the Technical University of Vienna *On the applicability of secret sharing cryptography in secure cloud services*

Ani Bicaku master thesis at the Carinthia University of Applied Sciences *Advanced Logging for Multilayer Cloud Compliance Supporting Assurance of Critical Infrastructure Cloud Services.*

(Co-)Instructed Bachelor Theses

Bozic Silvia bachelor thesis at the University of Applied Sciences Burgenland *A Comparison of Single Cloud, Multi Cloud, and Federated Cloud Architectures*

Roland Pellegrini bachelor thesis at the University of Applied Sciences Burgenland, part I.: *Einsatz einer nachrichtenorientierten Middleware im Treasury IT-Bereich einer Bank* part II.: *Einsatz von Raspberry Pi basierenden Ersatzarbeitsplätzen in einem Finanzinstitut*

Jelena Jozic bachelor thesis at the University of Applied Sciences Burgenland, part I.: *Scrum in der Softwareentwicklung – Einsatz und Alternativen* part II.: *Der Einsatz von Geoinformationssoftware in Kommunen - Die Software „GeoDesktop“ als Fallbeispiel*

Bibliography

[AAD12] Marco Anisetti, Claudio Agostino Ardagna, and Ernesto Damiani. A low-cost security certification scheme for evolving services. In *2012 IEEE 19th International Conference on Web Services, Honolulu, HI, USA, June 24-29, 2012*, pages 122–129, 2012.

[AAD14] Marco Anisetti, Claudio A Ardagna, and Ernesto Damiani. A certification-based trust model for autonomic cloud computing systems. In *Cloud and Autonomic Computing (ICCAC), 2014 International Conference on*, pages 212–219. IEEE, 2014.

[AAD15a] Marco Anisetti, Claudio Agostino Ardagna, and Ernesto Damiani. A test-based incremental security certification scheme for cloud-based systems. In *2015 IEEE International Conference on Services Computing, SCC 2015, New York City, NY, USA, June 27 - July 2, 2015*, pages 736–741, 2015.

[AAD+15b] Marco Anisetti, Claudio Agostino Ardagna, Ernesto Damiani, Filippo Gaudenzi, and Roberto Veca. Toward security and performance certification of open stack. In *8th IEEE International Conference on Cloud Computing, CLOUD 2015, New York City, NY, USA, June 27 - July 2, 2015*, pages 564–571, 2015.

[AADV14] Claudio Agostino Ardagna, Rasool Asal, Ernesto Damiani, and Quang Hieu Vu. On the management of cloud non-functional properties: The cloud transparency toolkit. In *6th International Conference on New Technologies, Mobility and Security, NTMS 2014, Dubai, United Arab Emirates, March 30 - April 2, 2014*, pages 1–4, 2014.

[AADV15] Claudio Agostino Ardagna, Rasool Asal, Ernesto Damiani, and Quang Hieu Vu. From security to assurance in the cloud: A survey. *ACM Comput. Surv.*, 48(1):2, 2015.

[ABD+12] Aditya Auradkar, Chavdar Botev, Shirshanka Das, Dave De Maagd, Alex Feinberg, Phanindra Ganti, Lei Gao, Bhaskar Ghosh, Kishore Gopalakrishna, Brendan Harris, Joel Koshy, Kevin Krawez, Jay Kreps, Shi Lu,

Sunil Nagaraj, Neha Narkhede, Sasha Pachev, Igor Perisic, Lin Qiao, Tom Quiggle, Jun Rao, Bob Schulman, Abraham Sebastian, Oliver Seeliger, Adam Silberstein, Boris Shkolnik, Chinmay Soman, Roshan Sumbaly, Kapil Surlaker, Sajid Topiwala, Cuong Tran, Balaji Varadarajan, Jemiah Westerman, Zach White, David Zhang, and Jason Zhang. "data infrastructure at linkedin". In *IEEE 28th International Conference on Data Engineering (ICDE 2012), Washington, DC, USA (Arlington, Virginia), 1-5 April, 2012*, pages 1370–1381, 2012.

[ABdDP13] Giuseppe Aceto, Alessio Botta, Walter de Donato, and Antonio Pescapè. Cloud monitoring: A survey. *Computer Networks*, 57(9):2093–2115, 2013.

[AF08] R. Abassi and S. Guemara El Fatmi. Towards an automated firewall security policies validation process. In *2008 Third International Conference on Risks and Security of Internet and Systems*, pages 267–272, Oct 2008.

[AFG+10] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy H. Katz, Andy Konwinski, Gunho Lee, David A. Patterson, Ariel Rabkin, Ion Stoica, and Matei Zaharia. A view of cloud computing. *Commun. ACM*, 53(4):50–58, 2010.

[AHW17] Philipp M. Radl Aleksandar Hudic, Paul Smith and Edgar R. Weippl. Security assurance assessment methodology for hybrid clouds. *Computers & Security*, 2017.

[AKF+10] R. E. Assad, T. Katter, F. S. Ferraz, L. P. Ferreira, and S. R. L. Meira. Security quality assurance on web-based application through security requirements tests: Elaboration, execution and automation. In *2010 Fifth International Conference on Software Engineering Advances*, pages 272–277, Aug 2010.

[AKG16] S. Alshamrani, D. Kowalski, and L. Gasieniec. The impact of hierarchical structure on efficiency of cloud monitoring. In *2016 2nd International Conference on Cloud Computing Technologies and Applications (CloudTech)*, pages 40–46, May 2016.

[AMH15] A. Marzouk A. Maarouf, M. E. Hamlaoui and A. Haqiq. Combining multi-agent systems and mde approach for monitoring sla violations in the cloud computing. In *2015 International Conference on Cloud Technologies and Applications (CloudTech)*, pages 1–6, June 2015.

[APST05] Thomas E. Anderson, Larry L. Peterson, Scott Shenker, and Jonathan S. Turner. Overcoming the internet impasse through virtualization. *IEEE Computer*, 38(4):34–41, 2005.

[APT15] Rocco Aversa, Nicola Panza, and Luca Tasquier. An agent-based platform for cloud applications performance monitoring. In *Ninth International*

Conference on Complex, Intelligent, and Software Intensive Systems, CISIS 2015, Santa Catarina, Brazil, July 8-10, 2015, pages 535–540, 2015.

[ASAM12] A. Abdi, A. Souzani, M. Amirfakhri, and A. B. Moghadam. Using security metrics in software quality assurance process. In *6th International Symposium on Telecommunications (IST)*, pages 1099–1102, Nov 2012.

[AZ14] Cristina Alcaraz and Sherali Zeadally. Critical infrastructure protection: Requirements and challenges for the 21st century. *International Journal of Critical Infrastructure Protection*, (0):–, 2014.

[BB13] A. Bilbao and E. Bilbao. Measuring security. In *2013 47th International Carnahan Conference on Security Technology (ICCST)*, pages 1–5, Oct 2013.

[BBC+15] Aline Bousquet, Jérémy Briffaut, Eddy Caron, Eva María Domínguez, Javier Franco, Arnaud Lefray, Óscar López, Saioa Ros, Jonathan Rouzaud-Cornabas, Christian Toinard, and Mikel Uriarte. Enforcing security and assurance properties in cloud environment. In *8th IEEE/ACM International Conference on Utility and Cloud Computing, UCC 2015, Limassol, Cyprus, December 7-10, 2015*, pages 271–280, 2015.

[Bea02] Robert G Bea. Human and organizational factors in reliability assessment and management of offshore structures. *Risk Analysis*, 22(1):29–45, 2002.

[BFH+15] Roland Bless, Matthias Flittner, Jens Horneber, Aleksandar Hudic, David Hutchison, Christian Jung, Frank Pallas, Markus Schöller, Syed Noor ul Hassan Shirazi, Steven Simpson, and Paul Smith. Seccrit architectural framework 2.0, 2015.

[BFL+13] André Brinkmann, Christoph Fiehe, Anna Litvina, Ingo Lück, Lars Nagel, Krishnaprasad Narayanan, Florian Ostermair, and Wolfgang Thronicke. Scalable monitoring system for clouds. In *IEEE/ACM 6th International Conference on Utility and Cloud Computing, UCC 2013, Dresden, Germany, December 9-12, 2013*, pages 351–356, 2013.

[BGO+16] Brendan Burns, Brian Grant, David Oppenheimer, Eric Brewer, and John Wilkes. Borg, omega, and kubernetes. *Commun. ACM*, 59(5):50–57, April 2016.

[BKW14a] Marianne Busch, Nora Koch, and Martin Wirsing. Evaluation of engineering approaches in the secure software development life cycle. In *Engineering Secure Future Internet Services and Systems - Current Research*, pages 234–265. 2014.

[BKW14b] Marianne Busch, Nora Koch, and Martin Wirsing. Seceval: An evaluation framework for engineering secure systems. In *Modellierung 2014, 19.-21. März 2014, Wien, Österreich*, pages 337–352, 2014.

[BML+11] Robert B. Bohn, John Messina, Fang Liu, Jin Tong, and Jian Mao. NIST cloud computing reference architecture. In *World Congress on Services, SERVICES 2011, Washington, DC, USA, July 4-9, 2011*, pages 594–596, 2011.

[Bre15] Eric A. Brewer. Kubernetes and the path to cloud native. In *Proceedings of the Sixth ACM Symposium on Cloud Computing*, SoCC '15, pages 167–167, New York, NY, USA, 2015. ACM.

[BVG14] Sören Bleikertz, Carsten Vogel, and Thomas Groß. Cloud radar: near real-time detection of security failures in dynamic virtualized infrastructures. In *Proceedings of the 30th Annual Computer Security Applications Conference, ACSAC 2014, New Orleans, LA, USA, December 8-12, 2014*, pages 26–35, 2014.

[BYV08] Rajkumar Buyya, Chee Shin Yeo, and Srikumar Venugopal. Market-oriented cloud computing: Vision, hype, and reality for delivering IT services as computing utilities. In *10th IEEE International Conference on High Performance Computing and Communications, HPCC 2008, 25-27 Sept. 2008, Dalian, China*, pages 5–13, 2008.

[CBR15] Valentina Casola, Alessandra De Benedictis, and Massimiliano Rak. Security monitoring in the cloud: An sla-based approach. In *10th International Conference on Availability, Reliability and Security, ARES 2015, Toulouse, France, August 24-27, 2015*, pages 749–755, 2015.

[CdDM+15] L. Catarinucci, D. de Donno, L. Mainetti, L. Palano, L. Patrono, M. L. Stefanizzi, and L. Tarricone. An iot-aware architecture for smart healthcare systems. *IEEE Internet of Things Journal*, 2(6):515–526, Dec 2015.

[CDZM13] Stelvio Cimato, Ernesto Damiani, Francesco Zavatarelli, and Renato Menicocci. Towards the certification of cloud services. In *IEEE Ninth World Congress on Services, SERVICES 2013, Santa Clara, CA, USA, June 28 - July 3, 2013*, pages 92–97, 2013.

[Ciu16] Augusto Ciuffoletti. Application level interface for a cloud monitoring service. *Computer Standards & Interfaces*, 46:15–22, 2016.

[CLLT13] Eddy Caron, Anh Dung Le, Arnaud Lefray, and Christian Toinard. Definition of security metrics for the cloud computing and security-aware virtual machine placement algorithms. In *2013 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, CyberC 2013, Beijing, China, October 10-12, 2013*, pages 125–131, 2013.

[Cou16] PCI Security Standards Council. Payment Card Industry Data Security Standard. https://www.pcisecuritystandards.org/, last accessed: 2016/03/25, 2016.

[CPL+97] Andrew A. Chien, Scott Pakin, Mario Lauria, Matt Buchanan, Kay Hane, Louis A. Giannini, and Jane Prusakova. High performance virtual machines (hpvm's): Clusters with supercomputing api's and performance. In *Proceedings of the Eighth SIAM Conference on Parallel Processing for Scientific Computing, PPSC 1997, March 14-17, 1997, Hyatt Regency Minneapolis on Nicollel Mall Hotel, Minneapolis, Minnesota, USA*, 1997.

[Cre02] John W Creswell. Educational research: Planning, conducting, and evaluating quantitative. *New Jersey: Upper Saddle River*, 2002.

[CRV10] A. Calvi, S. Ranise, and L. Vigano. Automated validation of security-sensitive web services specified in bpel and rbac. In *2010 12th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*, pages 456–464, Sept 2010.

[dCRdSG+14] Guilherme da Cunha Rodrigues, Glederson Lessa dos Santos, Vinicius Tavares Guimaraes, Lisandro Zambenedetti Granville, and Liane Margarida Rockenbach Tarouco. An architecture to evaluate scalability, adaptability and accuracy in cloud monitoring systems. In *The International Conference on Information Networking 2014, ICOIN 2014, Phuket, Thailand, February 10-12, 2014*, pages 46–51, 2014.

[Del15] Deloitte. Journey to Cloud 10 Questions. Technical report, Deloitte, 05 2015.

[Deo74] Narsingh Deo. *Graph Theory with Applications to Engineering and Computer Science (Prentice Hall Series in Automatic Computation)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1974.

[Din13] Wen Ding. Study of smart warehouse management system based on the iot. In *intelligence computation and evolutionary computation*, pages 203–207. Springer, 2013.

[DMM+12] Idilio Drago, Marco Mellia, Maurizio M. Munafò, Anna Sperotto, Ramin Sadre, and Aiko Pras. Inside dropbox: understanding personal cloud storage services. In *Proceedings of the 12th ACM SIGCOMM Internet Measurement Conference, IMC '12, Boston, MA, USA, November 14-16, 2012*, pages 481–494, 2012.

[Dou16] Lynne Doughtie. KPMG LLP Transparency report. Technical report, KPMG, 05 2016.

[DPW13] Bob Duncan, David J. Pym, and Mark Whittington. Developing a conceptual framework for cloud security assurance. In *IEEE 5th International Conference on Cloud Computing Technology and Science, CloudCom 2013, Bristol, United Kingdom, December 2-5, 2013, Volume 2*, pages 120–125, 2013.

[DW15] Bob Duncan and Mark Whittington. The importance of proper measurement for a cloud security assurance model. In *7th IEEE International Conference on Cloud Computing Technology and Science, CloudCom 2015, Vancouver, BC, Canada, November 30 - Dec. 3, 2015*, pages 517–522, 2015.

[EFL+99] Robert J. Ellison, David A. Fisher, Richard C. Linger, Howard F. Lipson, Thomas A. Longstaff, and Nancy R. Mead. Survivability: Protecting your critical systems. *IEEE Internet Computing*, 3(6):55–63, 1999.

[EY15] Ernst and Young. Global Information Security Survey - A life sciences perspective. Technical report, Ernst and Young, 11 2015.

[EY16] Ernst and Young. Governing the cloud - Drive innovation and empower your workforce through responsible adoption of the cloud. Technical report, Ernst and Young, 02 2016.

[FEH+14] Kaniz Fatema, Vincent C. Emeakaroha, Philip D. Healy, John P. Morrison, and Theo Lynn. A survey of cloud monitoring tools: Taxonomy, capabilities and objectives. *J. Parallel Distrib. Comput.*, 74(10):2918–2933, 2014.

[FIvL+99] Ian T. Foster, Joseph A. Insley, Gregor von Laszewski, Carl Kesselman, and Marcus Thiébaux. Distance visualization: Data exploration on the grid. *IEEE Computer*, 32(12):36–43, 1999.

[FK99] Ian Foster and Carl Kesselman, editors. *The Grid: Blueprint for a New Computing Infrastructure*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999.

[FKT01] Ian Foster, Carl Kesselman, and Steven Tuecke. The anatomy of the grid: Enabling scalable virtual organizations. *Int. J. High Perform. Comput. Appl.*, 15(3):200–222, August 2001.

[FLR+13] C. Fehling, F. Leymann, S.T. Ruehl, M. Rudek, and S. Verclas. Service Migration Patterns – Decision Support and Best Practices for the Migration of Existing Service-Based Applications to Cloud Environments. In *Service-Oriented Computing and Applications (SOCA), 2013 IEEE 6th International Conference on*, pages 9–16, Dec 2013.

[fSC05] International Organization for Standardization and International Electrotechnical Commission. Information technology – Security techniques – Information security management systems – Requirements. Standard, July 2005.

[fSC08a] International Organization for Standardization and International Electrotechnical Commission. Information technology security techniques — evaluation criteria for it security - part 3: Security assurance components. Standard, July 2008.

[fSC08b] International Organization for Standardization and International Electrotechnical Commission. Information technology — security techniques — methodology for it security evaluation. Standard, July 2008.

[fSC10] International Organization for Standardization and International Electrotechnical Commission. Information technology — security techniques — security assessment of operational systems. Standard, July 2010.

[fSC13] International Organization for Standardization and International Electrotechnical Commission. Information technology – Security techniques – Code of practice for information security controls. Standard, July 2013.

[fSC14] International Organization for Standardization and International Electrotechnical Commission. Iso/iec 27018 - information technology – security techniques – code of practice for protection of personally identifiable information (pii) in public clouds acting as pii processors. Standard, July 2014.

[fSC15] International Organization for Standardization and International Electrotechnical Commission. Iso/iec 27017 - information technology – security techniques – code of practice for information security controls based on iso/iec 27002 for cloud services. Standard, December 2015.

[fSC16] International Organization for Standardization and International Electrotechnical Commission. Iso/iec 19086-1:2016 - information technology – cloud computing – service level agreement (sla) framework. Standard, December 2016.

[GBD+16] F. R. Golra, A. Beugnard, F. Dagnat, S. Guerin, and C. Guychard. Continuous requirements engineering using model federation. In *2016 IEEE 24th International Requirements Engineering Conference (RE)*, pages 347–352, Sept 2016.

[GGGS11] Jesus Luna Garcia, Hamza Ghani, Daniel Germanus, and Neeraj Suri. A security metrics framework for the cloud. In *SECRYPT 2011 - Proceedings of the International Conference on Security and Cryptography, Seville, Spain, 18 - 21 July, 2011, SECRYPT is part of ICETE - The International*

Joint Conference on e-Business and Telecommunications, pages 245–250, 2011.

[GJ81] John E Gaffney Jr. Metrics in software quality assurance. In *Proceedings of the ACM'81 conference*, pages 126–130. ACM, 1981.

[GMM11] Javier González, Antonio Muñoz, and Antonio Maña. Multi-layer monitoring for cloud computing. In *13th IEEE International Symposium on High-Assurance Systems Engineering, HASE 2011, Boca Raton, FL, USA, November 10-12, 2011*, pages 291–298, 2011.

[GQ88] Louis C. Guillou and Jean-Jacques Quisquater. A practical zero-knowledge protocol fitted to security microprocessor minimizing both transmission and memory. In *Advances in Cryptology - EUROCRYPT '88, Workshop on the Theory and Application of of Cryptographic Techniques, Davos, Switzerland, May 25-27, 1988, Proceedings*, pages 123–128, 1988.

[Gro14] Thomas R. Groß. Efficient certification and zero-knowledge proofs of knowledge on infrastructure topology graphs. In *Proceedings of the 6th edition of the ACM Workshop on Cloud Computing Security, CCSW '14, Scottsdale, Arizona, USA, November 7, 2014*, pages 69–80, 2014.

[Hay10] Lance Hayden. *IT Security Metrics: A Practical Framework for Measuring Security & Protecting Data*. McGraw-Hill Education Group, 1st edition, 2010.

[Her02] Debra S Herrmann. *Using the Common Criteria for IT security evaluation*. CRC Press, 2002.

[HF10] Jez Humble and David Farley. *Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation (Adobe Reader)*. Pearson Education, 2010.

[HFL+16] Aleksandar Hudic, Matthias Flittner, Thomas Loruenser, Philipp Radl, and Roland Bless. Towards a unified secure cloud service development and deployment life-cycle. In *2016 International Conference on Availability, Reliability and Security ARES Salzburg, Austria,*, 2016.

[HGR+14] T. M. Hesse, S. Gärtner, T. Roehm, B. Paech, K. Schneider, and B. Bruegge. Semiautomatic security requirements engineering and evolution using decision documentation, heuristics, and user monitoring. In *Evolving Security and Privacy Requirements Engineering (ESPRE), 2014 IEEE 1st Workshop on*, pages 1–6, Aug 2014.

[HHT+14] Aleksandar Hudic, Thomas Hecht, Markus Tauber, Andreas Mauthe, and Santiago Cáceres Elvira. Towards continuous cloud service assurance for critical infrastructure IT. In *2014 International Conference on Future*

Internet of Things and Cloud, FiCloud 2014, Barcelona, Spain, August 27-29, 2014, pages 175–182, 2014.

[HL06] Michael Howard and Steve Lipner. *The security development lifecycle*, volume 8. Microsoft Press Redmond, 2006.

[HLMN08] C. Haley, R. Laney, J. Moffett, and B. Nuseibeh. Security requirements engineering: A framework for representation and analysis. *IEEE Transactions on Software Engineering*, 34(1):133–153, Jan 2008.

[HMC+14] H. He, Z. Ma, H. Chen, C. Y. Yeh, and W. Shao. An aspect-oriented approach to sla-driven monitoring multi-tenant cloud application. In *2014 IEEE 7th International Conference on Cloud Computing*, pages 857–864, June 2014.

[HS15] Kuo-Chan Huang and Bo-Jyun Shen. Service deployment strategies for efficient execution of composite saas applications on cloud platform. *Journal of Systems and Software*, 107:127–141, 2015.

[HSHJ08] Thomas Heyman, Riccardo Scandariato, Christophe Huygens, and Wouter Joosen. Using security patterns to combine security metrics. In *Proceedings of the The Third International Conference on Availability, Reliability and Security, ARES 2008, March 4-7, 2008, Technical University of Catalonia, Barcelona , Spain*, pages 1156–1163, 2008.

[HTL+14] Aleksandar Hudic, Markus Tauber, Thomas Lorünser, Maria Krotsiani, George Spanoudakis, Andreas Mauthe, and Edgar R. Weippl. A multilayer and multitenant cloud assurance evaluation methodology. In *IEEE 6th International Conference on Cloud Computing Technology and Science, CloudCom 2014, Singapore, December 15-18, 2014*, pages 386–393, 2014.

[HYC16] Yan Hu, Jun Yan, and Kim-Kwang Raymond Choo. PEDAL: a dynamic analysis tool for efficient concurrency bug reproduction in big data environment. *Cluster Computing*, 19(1):153–166, 2016.

[IIOW16] Umar Mukhtar Ismail, Shareeful Islam, Moussa Ouedraogo, and Edgar R. Weippl. A framework for security transparency in cloud computing. *Future Internet*, 8(1), 2016.

[Jaq07] Andrew Jaquith. *Security Metrics: Replacing Fear, Uncertainty, and Doubt*. Addison-Wesley Professional, 2007.

[Jaz14] N. Jazdi. Cyber physical systems in the context of industry 4.0. In *2014 IEEE International Conference on Automation, Quality and Testing, Robotics*, pages 1–4, May 2014.

[KGS10] Ali Khajeh-Hosseini, David Greenwood, and Ian Sommerville. Cloud migration: A case study of migrating an enterprise IT system to iaas. In *IEEE International Conference on Cloud Computing, CLOUD 2010, Miami, FL, USA, 5-10 July, 2010*, pages 450–457, 2010.

[KH13] Safwan Mahmud Khan and Kevin W. Hamlen. Computation certification as a service in the cloud. In *13th IEEE/ACM International Symposium on Cluster, Cloud, and Grid Computing, CCGrid 2013, Delft, Netherlands, May 13-16, 2013*, pages 434–441, 2013.

[KHGS10] A. Khajeh-Hosseini, D. Greenwood, and I. Sommerville. Cloud Migration: A Case Study of Migrating an Enterprise IT System to IaaS. In *Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on*, pages 450–457, July 2010.

[KHSBT11] A. Khajeh-Hosseini, I. Sommerville, J. Bogaerts, and P. Teregowda. Decision Support Tools for Cloud Migration in the Enterprise. In *Cloud Computing (CLOUD), 2011 IEEE International Conference on*, pages 541–548, July 2011.

[KJM+11] R. K. L. Ko, P. Jagadpramana, M. Mowbray, S. Pearson, M. Kirchberg, Q. Liang, and B. S. Lee. Trustcloud: A framework for accountability and trust in cloud computing. In *2011 IEEE World Congress on Services*, pages 584–588, July 2011.

[KM11] S. Kaisler and W.H. Money. Service Migration in a Cloud Architecture. In *System Sciences (HICSS), 2011 44th Hawaii International Conference on*, pages 1–10, Jan 2011.

[Kot04] Chakravanti Rajagopalachari Kothari. *Research methodology: Methods and techniques*. New Age International, 2004.

[KPR09] B.R. Kandukuri, V.R. Paturi, and A. Rakshit. Cloud Security Issues. In *Services Computing, 2009. SCC '09. IEEE International Conference on*, pages 517–520, Sept 2009.

[KPSD13] Igor Kotenko, Olga Polubelova, Igor Saenko, and Elena Doynikova. The ontology of metrics for security evaluation and decision support in siem systems. In *Availability, Reliability and Security (ARES), 2013 Eighth International Conference on*, pages 638–645. IEEE, 2013.

[Krö08] Wolfgang Kröger. Critical infrastructures at risk: A need for a new conceptual approach and extended analytical tools. *Rel. Eng. & Sys. Safety*, 93(12):1781–1787, 2008.

[KS14] Maria Krotsiani and George Spanoudakis. Continuous certification of non-repudiation in cloud storage services. In *13th IEEE International*

Conference on Trust, Security and Privacy in Computing and Communications, TrustCom 2014, Beijing, China, September 24-26, 2014, pages 921–928, 2014.

[KSCB03] Robin Koh, Edmund W Schuster, Indy Chackrabarti, and Attilio Bellman. Securing the pharmaceutical supply chain. *White Paper, Auto-ID Labs, Massachusetts Institute of Technology*, pages 1–19, 2003.

[KSM13] Maria Krotsiani, George Spanoudakis, and Khaled Mahbub. Incremental certification of cloud services. In *SECURWARE 2013-7th International Conference on Emerging Security Information, Systems and Technologies*, pages 72–80, 2013.

[KSM14] Spyros Katopodis, George Spanoudakis, and Khaled Mahbub. Towards hybrid cloud service certification models. In *IEEE International Conference on Services Computing, SCC 2014, Anchorage, AK, USA, June 27 - July 2, 2014*, pages 394–399, 2014.

[Lat86] Donald C Latham. Department of defense trusted computer system evaluation criteria. *Department of Defense*, 1986.

[LBRE14] Rafael Mira De Oliveira Libardi, Marcos Vinicius Naves Bedo, Stephan Reiff-Marganiec, and Júlio Cezar Estrella. MSSF: A step towards user-friendly multi-cloud data dispersal. In *2014 IEEE 7th International Conference on Cloud Computing, Anchorage, AK, USA, June 27 - July 2, 2014*, pages 952–953, 2014.

[LCTCY05] P. Lei, F. Claret-Tournier, C. Chatwin, and R. Young. A secure mobile track and trace system for anti-counterfeiting. In *2005 IEEE International Conference on e-Technology, e-Commerce and e-Service*, pages 686–689, March 2005.

[LHS15] Thomas Lorünser, Andreas Happe, and Daniel Slamanig. ARCHISTAR: towards secure and robust cloud based data sharing. In *7th IEEE International Conference on Cloud Computing Technology and Science, CloudCom 2015, Vancouver, BC, Canada, November 30 - Dec. 3, 2015*, pages 371–378, 2015.

[Lin16] David S. Linthicum. Emerging hybrid cloud patterns. *IEEE Cloud Computing*, 3(1):88–91, 2016.

[Lip04] Steven B. Lipner. The trustworthy computing security development lifecycle. In *20th Annual Computer Security Applications Conference (ACSAC 2004), 6-10 December 2004, Tucson, AZ, USA*, pages 2–13, 2004.

[Lip05] Steven B. Lipner. Building more secure commercial software: The trustworthy computing security development lifecycle. In *INFORMATIK 2005 - Informatik LIVE! Band 1, Beiträge der 35. Jahrestagung der Gesellschaft für Informatik e.V. (GI), Bonn, 19. bis 22. September 2005*, pages 21–28, 2005.

[Lip15] Steven B. Lipner. Security assurance. *Commun. ACM*, 58(11):24–26, October 2015.

[LIS16] Y. Lin, P. Ieromonachou, and W. Sun. Smart manufacturing and supply chain management. In *2016 International Conference on Logistics, Informatics and Service Sciences (LISS)*, pages 1–5, July 2016.

[LMB09] Federica Lince, DL Marchisio, and AA Barresi. Smart mixers and reactors for the production of pharmaceutical nanoparticles: Proof of concept. *Chemical Engineering Research and Design*, 87(4):543–549, 2009.

[LTSS15] Sebastian Lins, Scott Thiebes, Stephan Schneider, and Ali Sunyaev. What is really going on at your cloud service provider? creating trustworthy certifications by continuous auditing. In *48th Hawaii International Conference on System Sciences, HICSS 2015, Kauai, Hawaii, USA, January 5-8, 2015*, pages 5352–5361, 2015.

[MC12] Ben Martini and Kim-Kwang Raymond Choo. An integrated conceptual digital forensic framework for cloud computing. *Digital Investigation*, 9(2):71–80, 2012.

[MC13] Ben Martini and Kim-Kwang Raymond Choo. Cloud storage forensics: owncloud as a case study. *Digital Investigation*, 10(4):287–299, 2013.

[MC14] Ben Martini and Kim-Kwang Raymond Choo. Distributed filesystem forensics: Xtreemfs as a case study. *Digital Investigation*, 11(4):295–313, 2014.

[Mea12] N. R. Mead. Measuring the software security requirements engineering process. In *2012 IEEE 36th Annual Computer Software and Applications Conference Workshops*, pages 583–588, July 2012.

[MFB16] Silvia Balaban Matthias Flittner and Roland Bless. Cloudinspector: A transparency-as-a-service solution for legal issues in cloud computing. *IEEE 2nd Workshop on Legal and Technical Issues in Cloud Computing and Cloud-Supported Internet of Things*, April 2016.

[MFMP06] Daniel Mellado, Eduardo Fernandez-Medina, and Mario Piattini. Applying a security requirements engineering process. In *Computer Security–ESORICS 2006*, pages 192–206. Springer, 2006.

[MFMP07] Daniel Mellado, Eduardo Fernández-Medina, and Mario Piattini. A common criteria based security requirements engineering process for the development of secure information systems. *Computer standards & interfaces*, 29(2):244–253, 2007.

[MG11] Peter Mell and Tim Grance. The nist definition of cloud computing. 2011.

[MLN+12] Matt Massie, Bernard Li, Brad Nicholes, Vladimir Vuksan, Robert Alexander, Jeff Buchbinder, Frederiko Costa, Alex Dean, Dave Josephsen, Peter Phaal, et al. *Monitoring with Ganglia*. " O'Reilly Media, Inc.", 2012.

[MNP+11] Philippe Massonet, Syed Naqvi, Christophe Ponsard, Joseph Latanicki, Benny Rochwerger, and Massimo Villari. A monitoring and audit logging architecture for data location compliance in federated cloud infrastructures. In *25th IEEE International Symposium on Parallel and Distributed Processing, IPDPS 2011, Anchorage, Alaska, USA, 16-20 May 2011 - Workshop Proceedings*, pages 1510–1517, 2011.

[MP16] S. M. Muthukrishnan and S. Palaniappan. Security metrics maturity model for operational security. In *2016 IEEE Symposium on Computer Applications Industrial Electronics (ISCAIE)*, pages 101–106, May 2016.

[MZS+14] Javier Diaz Montes, Mengsong Zou, Rahul Singh, Shu Tao, and Manish Parashar. Data-driven workflows in multi-cloud marketplaces. In *2014 IEEE 7th International Conference on Cloud Computing, Anchorage, AK, USA, June 27 - July 2, 2014*, pages 168–175, 2014.

[NA12] Nist and Emmanuel Aroms. *NIST Special Publication 800-53 Revision 3 Recommended Security Controls for Federal Information Systems and Organizations*. CreateSpace, Paramount, CA, 2012.

[NBVS13] V. K. Naik, K. Beaty, N. Vogl, and J. Sanchez. Workload monitoring in hybrid clouds. In *Cloud Computing (CLOUD), 2013 IEEE Sixth International Conference on*, pages 816–822, June 2013.

[NCAW14] Mihir Nanavati, Patrick Colp, Bill Aiello, and Andrew Warfield. Cloud Security: A Gathering Storm. *Commun. ACM*, 57(5):70–79, May 2014.

[NSHS14] The An Binh Nguyen, Melanie Siebenhaar, Ronny Hans, and Ralf Steinmetz. Role-based templates for cloud monitoring. In *Proceedings of the 7th IEEE/ACM International Conference on Utility and Cloud Computing, UCC 2014, London, United Kingdom, December 8-11, 2014*, pages 242–250, 2014.

[Nug15] Y. Nugraha. Security assurance requirements engineering (stare) for trustworthy service level agreements. In *2015 IEEE 23rd International Requirements Engineering Conference (RE)*, pages 398–399, Aug 2015.

[OI15] Moussa Ouedraogo and Shareeful Islam. Towards the integration of security transparency in the modelling and design of cloud based systems. In *Advanced Information Systems Engineering Workshops - CAiSE 2015 International Workshops, Stockholm, Sweden, June 8-9, 2015, Proceedings*, pages 495–506, 2015.

[OWA16] OWASP. Application Security Verification Standard. `https://www.owasp.org/`, last accessed: 2016/03/25, 2016.

[Pah15a] Claus Pahl. Containerization and the paas cloud. *IEEE Cloud Computing*, 2(3):24–31, 2015.

[Pah15b] Claus Pahl. Containerization and the paas cloud. *IEEE Cloud Computing*, 2(3):24–31, 2015.

[Pat94] David A. Patterson. A case for NOW (networks-of-workstations). In *Hot Interconnects II - Symposium Record, Stanford, CA, USA, August 11-13, 1994*, pages 43–58, 1994.

[PC10] S. Pfleeger and R. Cunningham. Why measuring security is hard. *IEEE Security Privacy*, 8(4):46–54, July 2010.

[PESSGTL15] J. A. Perez-Espinoza, V. J. Sosa-Sosa, J. L. Gonzalez, and E. Tello-Leal. A distributed architecture for monitoring private clouds. In *2015 26th International Workshop on Database and Expert Systems Applications (DEXA)*, pages 186–190, Sept 2015.

[Pet14] D. Petcu. A taxonomy for sla-based monitoring of cloud security. In *2014 IEEE 38th Annual Computer Software and Applications Conference*, pages 640–641, July 2014.

[PHM+12] Fawaz Paraiso, Nicolas Haderer, Philippe Merle, Romain Rouvoy, and Lionel Seinturier. A federated multi-cloud paas infrastructure. In *2012 IEEE Fifth International Conference on Cloud Computing, Honolulu, HI, USA, June 24-29, 2012*, pages 392–399, 2012.

[Pig14] Richard Piggin. Are industrial control systems ready for the cloud? *International Journal of Critical Infrastructure Protection*, (0):–, 2014.

[PL15] Claus Pahl and Brian Lee. Containers and clusters for edge cloud architectures–a technology review. In *Future Internet of Things and Cloud (FiCloud), 2015 3rd International Conference on*, pages 379–386. IEEE, 2015.

[Pri15] PricewaterhouseCoopers. Building trust through assurance. Technical report, PricewaterhouseCoopers, 11 2015.

[Pri16] PricewaterhouseCoopers. Industry 4.0: Building the digital enterprise. Technical report, PricewaterhouseCoopers, 2016.

[PvdH06] Mike P. Papazoglou and Willem-Jan van den Heuvel. Service-oriented design and development methodology. *Int. J. Web Eng. Technol.*, 2(4):412–442, 2006.

[QC13a] Darren Quick and Kim-Kwang Raymond Choo. Digital droplets: Microsoft skydrive forensic data remnants. *Future Generation Comp. Syst.*, 29(6):1378–1394, 2013.

[QC13b] Darren Quick and Kim-Kwang Raymond Choo. Forensic collection of cloud storage data: Does the act of collection result in changes to the data or its metadata? *Digital Investigation*, 10(3):266–277, 2013.

[QC14] Darren Quick and Kim-Kwang Raymond Choo. Google drive: Forensic analysis of data remnants. *J. Netw. Comput. Appl.*, 40:179–193, April 2014.

[QPEM10] N. A. Qureshi, A. Perini, N. A. Ernst, and J. Mylopoulos. Towards a continuous requirements engineering framework for self-adaptive systems. In *2010 First International Workshop on Requirements@Run.Time*, pages 9–16, Sept 2010.

[Ran93] Kai Rannenberg. Recent development in information technology security evaluation-the need for evaluation criteria for multilateral security. In *Security and control of information technology in society*, pages 113–128, 1993.

[Ran14] Rajiv Ranjan. Streaming big data processing in datacenter clouds. *IEEE Cloud Computing*, 1(1):78–83, 2014.

[RBL+09] Benny Rochwerger, David Breitgand, Eliezer Levy, Alex Galis, Kenneth Nagin, Ignacio Martín Llorente, Rubén S. Montero, Yaron Wolfsthal, Erik Elmroth, Juan A. Cáceres, Muli Ben-Yehuda, Wolfgang Emmerich, and Fermín Galán. The reservoir model and architecture for open federated cloud computing. *IBM Journal of Research and Development*, 53(4):4, 2009.

[RGS+16] M. A. T. Rojas, N. M. Gonzalez, F. V. Sbampato, F. F. Redígolo, T. Carvalho, K. W. Ullah, M. Näslund, and A. S. Ahmed. A framework to orchestrate security sla lifecycle in cloud computing. In *2016 11th Iberian Conference on Information Systems and Technologies (CISTI)*, pages 1–7, June 2016.

[RGYC16] Nurul Hidayah Ab Rahman, William Bradley Glisson, Yanjiang Yang, and Kim-Kwang Raymond Choo. Forensic-by-design framework for cyber-physical cloud systems. *IEEE Cloud Computing*, 3(1):50–59, 2016.

[Rig17] RightScale. State of the cloud. Technical report, RightScale, 02 2017.

[RMR+16] E. Rios, W. Mallouli, M. Rak, V. Casola, and A. M. Ortiz. Sla-driven monitoring of multi-cloud application components using the musa framework. In *2016 IEEE 36th International Conference on Distributed Computing Systems Workshops (ICDCSW)*, pages 55–60, June 2016.

[RPK01] S. M. Rinaldi, J. P. Peerenboom, and T. K. Kelly. Identifying, understanding, and analyzing critical infrastructure interdependencies. *IEEE Control Systems*, 21(6):11–25, Dec 2001.

[RST01] Ronald L. Rivest, Adi Shamir, and Yael Tauman. How to leak a secret. In *Advances in Cryptology - ASIACRYPT 2001, 7th International Conference on the Theory and Application of Cryptology and Information Security, Gold Coast, Australia, December 9-13, 2001, Proceedings*, pages 552–565, 2001.

[RVM+11] Massimiliano Rak, Salvatore Venticinque, Tamás Máhr, Gorka Echevarria, and Gorka Esnal. Cloud application monitoring: The mosaic approach. In *IEEE 3rd International Conference on Cloud Computing Technology and Science, CloudCom 2011, Athens, Greece, November 29 - December 1, 2011*, pages 758–763, 2011.

[SA09] R. M. Savola and H. Abie. Development of security metrics for a distributed messaging system. In *Application of Information and Communication Technologies, 2009. AICT 2009. International Conference on*, pages 1–6, Oct 2009.

[Sal16] Diana Salazar. Cloud Security Framework Audit Methods. Technical report, SANS, 02 2016.

[Sav07] R. Savola. Towards a security metrics taxonomy for the information and communication technology industry. In *Software Engineering Advances, 2007. ICSEA 2007. International Conference on*, pages 60–60, Aug 2007.

[SB02] Ken Schwaber and Mike Beedle. *Agile software development with Scrum*, volume 1. Prentice Hall Upper Saddle River, 2002.

[SDM12] George Spanoudakis, Ernesto Damiani, and Antonio Maña. Certifying services in cloud: The case for a hybrid, incremental and multi-layer approach. In *14th International IEEE Symposium on High-Assurance Systems Engineering, HASE 2012, Omaha, NE, USA, October 25-27, 2012*, pages 175–176, 2012.

[SFM+16] K. Shaukat, A. Faisal, R. Masood, A. Usman, and U. Shaukat. Security quality assurance through penetration testing. In *2016 19th International Multi-Topic Conference (INMIC)*, pages 1–6, Dec 2016.

[SGK+16] Jose M. Such, Antonios Gouglidis, William Knowles, Gaurav Misra, and Awais Rashid. Information assurance techniques: Perceived cost effectiveness. *Computers & Security*, 60:117 – 133, 2016.

[SJL+11] Kun Sun, Sushil Jajodia, Jason Li, Yi Cheng, Wei Tang, and Anoop Singhal. Automatic security analysis using security metrics. In *MILITARY COMMUNICATIONS CONFERENCE, 2011-MILCOM 2011*, pages 1207–1212. IEEE, 2011.

[SK11] S. Subashini and V. Kavitha. A survey on security issues in service delivery models of cloud computing. *Journal of Network and Computer Applications*, 34(1):1–11, 2011.

[SKZ11] Yashaswi Singh, Farah Kandah, and Weiyi Zhang. A secured cost-effective multi-cloud storage in cloud computing. In *Computer Communications Workshops (INFOCOM WKSHPS), 2011 IEEE Conference on*, pages 619–624. IEEE, 2011.

[SL16] S. Singh and Y. Liu. A cloud service architecture for analyzing big monitoring data. *Tsinghua Science and Technology*, 21(1):55–70, Feb 2016.

[SLGS14] Stephan Schneider, Jens Lansing, Fangjian Gao, and Ali Sunyaev. A taxonomic perspective on certification schemes: Development of a taxonomy for cloud service certification criteria. In *47th Hawaii International Conference on System Sciences, HICSS 2014, Waikoloa, HI, USA, January 6-9, 2014*, pages 4998–5007, 2014.

[SME15] P. Martin S. Moustafa, K. Elgazzar and M. Elsayed. Slam: Sla monitoring framework for federated cloud services. In *2015 IEEE/ACM 8th International Conference on Utility and Cloud Computing (UCC)*, pages 506–511, Dec 2015.

[SN07] B. Steffen and P. Narayan. Full life-cycle support for end-to-end processes. *Computer*, 40(11):64–73, Nov 2007.

[SS13] Ali Sunyaev and Stephan Schneider. Cloud services certification. *Commun. ACM*, 56(2):33–36, 2013.

[SSB+95] Thomas L. Sterling, Daniel Savarese, Donald J. Becker, John E. Dorband, Udaya A. Ranawake, and Charles V. Packer. BEOWULF: A parallel workstation for scientific computation. In *Proceedings of the 1995 International Conference on Parallel Processing, Urbana-Champain, Illinois, USA, August 14-18, 1995. Volume I: Architecture.*, pages 11–14, 1995.

[SSE+15] Reijo M. Savola, Pekka Savolainen, Antti Evesti, Habtamu Abie, and Markus Sihvonen. Risk-driven security metrics development for an e-health iot application. In *2015 Information Security for South Africa, ISSA 2015, Johannesburg, South Africa, August 12-13, 2015*, pages 1–6, 2015.

[Ste46] Stanley Smith Stevens. On the theory of scales of measurement, 1946.

[SWWM10] Jin Shao, Hao Wei, Qianxiang Wang, and Hong Mei. A runtime model based monitoring approach for cloud. In *IEEE International Conference on Cloud Computing, CLOUD 2010, Miami, FL, USA, 5-10 July, 2010*, pages 313–320, 2010.

[TJA10] Hassan Takabi, James B. D. Joshi, and Gail-Joon Ahn. Security and privacy challenges in cloud computing environments. *IEEE Security & Privacy*, 8(6):24–31, 2010.

[TTS+14] Ankit Toshniwal, Siddarth Taneja, Amit Shukla, Karthik Ramasamy, Jignesh M. Patel, Sanjeev Kulkarni, Jason Jackson, Krishna Gade, Maosong Fu, Jake Donham, Nikunj Bhagat, Sailesh Mittal, and Dmitriy Ryaboy. Storm@twitter. In *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*, SIGMOD '14, pages 147–156, New York, NY, USA, 2014. ACM.

[TVS02] Andrew S Tanenbaum and Maarten Van Steen. *Distributed systems: principles and paradigms*, volume 2. Prentice hall Englewood Cliffs, 2002.

[VP14] R. Vaarandi and M. Pihelgas. Using security logs for collecting and reporting technical security metrics. In *Military Communications Conference (MILCOM), 2014 IEEE*, pages 294–299, Oct 2014.

[vRBV03] Robbert van Renesse, Kenneth P. Birman, and Werner Vogels. Astrolabe: A robust and scalable technology for distributed system monitoring, management, and data mining. *ACM Trans. Comput. Syst.*, 21(2):164–206, 2003.

[VSR14] Ricardo Villalón-Fonseca, Braulio J. Solano-Rojas, and Gabriela Marín Raventós. An applied methodology for information security and assurance: A study case for cloud computing. In *9th International Conference for Internet Technology and Secured Transactions, ICITST 2014, London, United Kingdom, December 8-10, 2014*, pages 432–439, 2014.

[WB14] Jonathan Stuart Ward and Adam Barker. Observing the clouds: a survey and taxonomy of cloud monitoring. *Journal of Cloud Computing*, 3(1):24, 2014.

[WBTY11] Philipp Wieder, Joe M Butler, Wolfgang Theilmann, and Ramin Yahyapour. *Service level agreements for cloud computing.* Springer Science & Business Media, 2011.

[WCW+13] Cong Wang, Sherman S. M. Chow, Qian Wang, Kui Ren, and Wenjing Lou. Privacy-preserving public auditing for secure cloud storage. *IEEE Trans. Computers*, 62(2):362–375, 2013.

[WG14] M. Whaiduzzaman and A. Gani. Measuring security for cloud service provider: A third party approach. In *2013 International Conference on Electrical Information and Communication Technology (EICT)*, pages 1–6, Feb 2014.

[WHM+15] Christian Wagner, Aleksandar Hudic, Silia Maksuti, Markus Tauber, and Frank Pallas. Impact of critical infrastructure requirements on service migration guidelines to the cloud. In *3rd International Conference on Future Internet of Things and Cloud, FiCloud 2015, Rome, Italy, August 24-26, 2015*, pages 1–8, 2015.

[WKS+15] Guozhang Wang, Joel Koshy, Sriram Subramanian, Kartik Paramasivam, Mammad Zadeh, Neha Narkhede, Jun Rao, Jay Kreps, and Joe Stein. Building a replicated logging system with apache kafka. *PVLDB*, 8(12):1654–1655, 2015.

[WLB09] Jian Wu, Qianhui Liang, and Elisa Bertino. Improving scalability of software cloud for composite web services. In *IEEE International Conference on Cloud Computing, CLOUD 2009, Bangalore, India, 21-25 September, 2009*, pages 143–146, 2009.

[WLL12] Boyang Wang, Baochun Li, and Hui Li. Oruta: Privacy-preserving public auditing for shared data in the cloud. In *2012 IEEE Fifth International Conference on Cloud Computing, Honolulu, HI, USA, June 24-29, 2012*, pages 295–302, 2012.

[WLL15] Boyang Wang, Baochun Li, and Hui Li. Panda: Public auditing for shared data with efficient user revocation in the cloud. *IEEE Trans. Services Computing*, 8(1):92–106, 2015.

[WRC+16] Y. Wang, S. Ram, F. Currim, E. Dantas, and L. A. Sabóia. A big data approach for smart transportation management on bus network. In *2016 IEEE International Smart Cities Conference (ISC2)*, pages 1–6, Sept 2016.

[WS13] Iryna Windhorst and Ali Sunyaev. Dynamic certification of cloud services. In *2013 International Conference on Availability, Reliability and Security, ARES 2013, Regensburg, Germany, September 2-6, 2013*, pages 412–417, 2013.

[WWRL10] Cong Wang, Qian Wang, Kui Ren, and Wenjing Lou. Privacy-preserving public auditing for data storage security in cloud computing. In *INFOCOM 2010. 29th IEEE International Conference on Computer Communications, Joint Conference of the IEEE Computer and Communications Societies, 15-19 March 2010, San Diego, CA, USA*, pages 525–533, 2010.

[XWP14] Teng Xu, James B. Wendt, and Miodrag Potkonjak. Security of iot systems: Design challenges and opportunities. In *Proceedings of the 2014 IEEE/ACM International Conference on Computer-Aided Design*, ICCAD '14, pages 417–423, Piscataway, NJ, USA, 2014. IEEE Press.

[YBP+06] Chee Shin Yeo, Rajkumar Buyya, Hossein Pourreza, M. Rasit Eskicioglu, Peter Graham, and Frank Sommers. Cluster computing: High-performance, high-availability, and high-throughput processing on a network of computers. In *Handbook of Nature-Inspired and Innovative Computing - Integrating Classical Models with Emerging Technologies*, pages 521–551. 2006.

[YK13] MM Younis A Younis and K Kifayat. Secure cloud computing for critical infrastructure: A survey. *Liverpool John Moores University, United Kingdom, Tech. Rep*, 2013.

[You16] Mazin Yousif. Hybrid clouds. *IEEE Cloud Computing*, 3(1):6–7, 2016.

[ZCF+10] Matei Zaharia, Mosharaf Chowdhury, Michael J. Franklin, Scott Shenker, and Ion Stoica. Spark: Cluster computing with working sets. In *2nd USENIX Workshop on Hot Topics in Cloud Computing, HotCloud'10, Boston, MA, USA, June 22, 2010*, 2010.

[ZL12] Dimitrios Zissis and Dimitrios Lekkas. Addressing cloud computing security issues. *Future Generation Computer Systems*, 28(3):583–592, 2012.

Publication List

Within this chapter a list of relevant publications that contain core scientific contributions in this thesis are introduced.

1. Hudic, A., Hecht, T., Tauber, M., Mauthe, A., & Elvira, S. C. (2014, August). Towards continuous cloud service assurance for critical infrastructure IT. In Future Internet of Things and Cloud (FiCloud), 2014 International Conference on (pp. 175-182). IEEE.

    - **A. Hudic - scientific and technical contributor: assurance evaluation methodology, assurance framework concept, state of the art evaluation**
    - T. Hecht - technical contributor: technical discussion and contribution towards potential use cases
    - M. Tauber - scientific reviewer
    - A. Mauthe - scientific reviewer
    - S. C. Elvira - technical reviewer: technical discussion of potential use cases

2. Wagner, C., Hudic, A., Maksuti, S., Tauber, M., & Pallas, F. (2015, August). Impact of Critical Infrastructure Requirements on Service Migration Guidelines to the Cloud. In Future Internet of Things and Cloud (FiCloud), 2015 3rd International Conference on (pp. 1-8). IEEE.

    - C. Wagner - technical reviewer and contributor: survey analysis, secure cloud migration taxonomy
    - **A. Hudic - scientific and technical contributor: design of the process-based information security guideline for cloud migration, survey analysis, evaluation of standards best practices and state of the art, secure cloud migration taxonomy**
    - S. Maksuti - technical reviewer and contributor: inserting survey results in to a analytic database
    - M. Tauber - scientific reviewer

- F. Pallas - technical reviewer

3. Hudic, A., Tauber, M., Lorunser, T., Krotsiani, M., Spanoudakis, G., Mauthe, A., & Weippl, E. R. (2014, December). A multi-layer and multitenant cloud assurance evaluation methodology. In Cloud Computing Technology and Science (CloudCom), 2014 IEEE 6th International Conference on (pp. 386-393). IEEE.

   - **A. Hudic - scientific and technical contributor: cloud assurance evaluation methodology, hierarchical abstraction model, assurance aggregation model, assurance algorithm, analytical evaluation, security properties profiling**
   - M. Tauber - scientific reviewer
   - T. Lorunser - scientific reviewer
   - M. Krotsiani - scientific contributor: integration of security requirements from CUMULUS project, elicitation of threats, section B the table and justifications
   - G. Spanoudakis - scientific reviewer and contributor: integration of security requirements from CUMULUS project, elicitation of threats, section B the table and justifications
   - E. Weippl - scientific reviewer

4. Hudic, A., Matthias F., Lorunser, T., Philipp R. and Roland B. (2016, August). Towards a Unified Secure Cloud Service Development and Deployment Life-cycle. In Availability, Reliability and Security (ARES), 2015 10th International Conference on (pp. 501-510). IEEE.

   - **A. Hudic - scientific and technical contributor: uniform secure design,develpoment and deployment life cycle, information security requirements analysis and engineering**
   - M. Flittner - scientific contributor: integration of Cloud inspector
   - T. Lorunser - technical reviewer
   - P. Radl - technical reviewer
   - R. Bless - technical reviewer

5. Hudic, A., Smith P. and Weippl E. Security Assessment Methodology for Multi-layered Clouds Elsevier Computers & Security Journal

   - **A. Hudic - scientific and technical contributor: security assessment methodology for multi-layerd clouds, empirical evaluation, implementation of the technical model, administration of cloud testbed environment, assurance assessment methodology - algorithms, aggregation protocol, security properties, assurance levels, assurance evaluation mode, protection profile, prototype implementation**

- P. Smith - scientific reviewer
- E. Weippl - scientific reviewer

Towards Continuous Cloud Service Assurance for Critical Infrastructure IT

The paper "Towards Continuous Cloud Service Assurance for Critical Infrastructure IT" was published on IEEE International Conference on Future Internet of Things and Cloud 2014 in Barcelona.

The paper can be found online under the following link: https://www.computer.org/csdl/proceedings/ficloud/2014/4357/00/4357a175.pdf

Towards Continuous Cloud Service Assurance for Critical Infrastructure IT

Aleksandar Hudic, Thomas Hecht, Markus Tauber
Austrian Institute of Technology
{alex.hudic, thomas.hecht, markus.tauber}@ait.ac.at

Andreas Mauthe
Lancaster University
a.mauthe@lancaster.ac.uk

Santiago Cáceres Elvira
ETRA Investigación y Desarrollo S.A.
scaceres.etra-id@grupoetra.com

***Abstract*—The momentum behind *Cloud Computing* has revolutionized how ICT services are provided, adopted and delivered. Features such as high scalability, fast provisioning, on demand resource availability makes it an attractive proposition for deploying complex and demanding systems. Clouds are also very suitable for deploying systems with unpredictable load patterns including *Critical infrastructure services*. Though, the major obstacle in hosting Critical infrastructures is often a lack of assurance. The transparency and flexibility offered by the Cloud, abstracts per definition over e.g. data placement, hardware, service migration. This makes it very hard to assure security properties. We present an investigation of assurance approaches, an analysis of their suitability for Critical Infrastructure Services being deployed in the Cloud and presents our approach.**

I. INTRODUCTION

Public utilities such as water, electricity, public transportation, health care system and telecommunication are vital assets of each society. Therefore, these assets are considered as the essential utility that drive economies and societies worldwide. Due to their crucial role, they are commonly referred in literature as Critical Infrastructure (CI) [1], [2], [3]. IT Systems used for managing CI require large resources, and hence CI providers often host their own infrastructure and possess own data centers.

A system defined by National Institute of Standards and Technology[1] (NIST), is a model for providing ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources, is more familiar under the term *Cloud Computing*. However, multitude of services offered by the Cloud makes it a viable proposition to embrace the Critical infrastructures in the Cloud environment (e.g. resilience to natural disasters, faster recovery in case of failure, redundancy, etc.), but it also results in new challenges as well (e.g. Loss of human-operated control for verifying security and privacy settings, week authentication and access control, denial of service, service failures, interference attacks, locality and legislative issues, data recovery issues, violation of service agreements, etc.). The very nature of cloud computing means that a service built on top of it comprises a multitude of heterogeneous components. The combination of these components may vary over time and administrative and/or geographical boundaries. This makes it hard to assure security properties of the deployed services – this is however of great importance of CI Systems.

We hence focus on investigating how existing assurance approaches can be applied to Cloud when being used for deploying CI services. Assurance in this context refers to guaranteeing security properties of a service which stores and process CI data inside the Cloud. A more detailed definition of assurance follows in the Section II.

This results into various research questions for assurance:

- How to derive a cloud service's overall assurance if individual assurance levels have to be aggregated?
- How to provide continuous assurance of a system?
- How to aggregate assurance levels across various legislative and administration boundaries?
- How to process assurance evaluation in an automated manner, and which guidelines exist?
- What are the issues related with Assurance for CI hosted in Cloud ecosystem?

In Section II we define assurance, outline our objectives, and illustrate the scope of our problem space. In Section III we show applied assurance as used within our research project. Section IV presents a comprehensive state-of-the-art evaluation and a discussion about shortcomings of exiting approaches in respect to our research questions and CI requirements. Finally we conclude the paper and present our future regarding the development of an assurance approach, for the given CI context, in section V.

II. ASSURANCE

Mechanism offered by each Cloud provider nowadays, for ensuring quality of service, are mainly based on Service Level Agreements (SLA). However, SLAs define mainly a predefined probability for delivering specific services in the Cloud environment. What is lacking in a SLA [4], is the assurance that measurable security & privacy properties & mechanisms are continuously met. For example, for data or information inside an ICT system, one of many challenges/requirements is a well-defined level of data confidentiality in order to maintain privacy across administrative and geographical borders. To ensure the data confidentiality the most easiest and intuitive approach would be to encrypt and to restrict the access. The user's data in order to reduce performance and processing costs is often stored unencrypted. The drawback of the approach, (i.e. leaving the data unprotected), is that it opens the possibility for significant data losses or exposures to unauthorized parties. Another example refers to deployment of virtual machines on the top of a Cloud's infrastructure layer, depending on the network and infrastructure components. Short outage of the only one component, regardless of the source of failure, regular firmware or software upgrades, migration on new virtualization stack, mitigates the possibility to continuously insure service

[1]NIST, http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf

978-1-4244-8396-9/10/$26.00

or information provisioning. In order to ensure that appropriate measures in the Cloud are met, we have to analyze and estimate each individual system components, services or actions.
Research directed towards this investigates if proper measures and actions have been undertaken to protect the data through its life cycle. This is known as Information Assurance. The USA's Department of Defense [5] defined information assurance as a measure that protects and defends information and information systems by ensuring their availability, integrity, authentication, confidentiality, and non-repudiation. This includes providing for restoration of information systems by incorporating protection, detection, and reaction capabilities.
For the purpose of this paper the overall assurance is observed in respect to the proposed architecture and it's dynamic and volatile properties. Therefore, in respect with [5], we extend the ***assurance*** definition as the volatile property of a dynamic ICT system which enables to quantify each individual component based upon the confidence to reliably determine integrity, availability and confidentiality of the data or the services that the system provides. Furthermore, we additionally distinguish the following two assurance elements: ***System assurance*** and ***Information assurance***, defined as it follows:

- System assurance defines the assurance of individual components of a system such as service, class, or a module, and their mutual correlations.
- Information assurance defines the assurance of the data governance in respect to a single element or component.

In order to achieve the overall aggregation of all entities in respect with their dynamic properties, the above mentioned assurance elements are classified per the following three dimensions: standalone entities that are able to produce output based on the incoming input, *component*; set of individual components compounded to deliver a service, *layer*; and connections between the individual components which mutually deliver information or service, *dependency*. Finally, our goal is to ensure the continuity in delivering assurance regardless if we are taking in to consideration a single element, a layer or a whole system. Hence, we consider dynamic properties as a crucial element for delivering continuous assurance. For achieving the continuous assurance, we tend to investigate the how often and in which intervals should we evaluate particular assurance properties. We are motivated by the work [6] for investigating intervals and the work [7], [8], [9]

III. ASSURANCE AND SECCRIT: A CASE STUDY

A. SECCRIT architecture

We investigate our research questions and objectives within the scope of the SEcure Cloud computing for Critical infrastructure IT (SECCRIT) project. The SECCRIT project's mission is to identify relevant legal frameworks and establishment of respective guidelines, provisioning of evidence and data protection for cloud services; understanding, assessing and managing risk associated with cloud environments; establishing best practice for secure cloud service implementations; and the demonstration of SECCRIT research and development results in real-world application scenarios. An important contribution of SECCRIT is to provide a reference architecture [10], depicted in Figure 1, for supporting the development of technical solution for the provisioning of evidence and data protection for cloud services [11]. The level based classification addresses

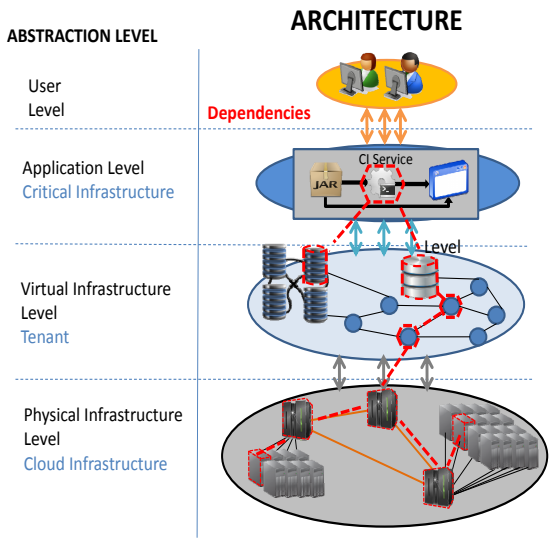

Fig. 1. Components and dependencies relevant for assuring application level security properties, illustrated in an architectural framework[10]. The left hand side of the figure depicts the abstraction levels of the architecture used for distinguishing individual assurance levels, which are additionally granulated through level components/elements.

directly the levels depicted on the left hand side of the Figure 1, and granularity is based on the individual components in particular level of the architecture.

B. Assurance in SECCRIT

The SECCRIT architecture, illustrated in Figure 1, is a structured approach for conducting our research on assurance for CI IT in the Cloud. Therefore, we also refer to the SECCRIT architecture[10] – which addresses the specific requirements of CI providers. Consequently within the scope of this work we propose a solution for addressing continuous assurance in Cloud ecosystem. The following set of properties & elements in line with the SECCRIT architecture, introduced in Figure 1, for assessing assurance should be considered:

- **Service components:**
  volatile elements: modules, classes and services;
  level: Application Level
- **Application components**
  volatile elements: API, frameworks, libraries;
  level: Application Level
- **Platform components**
  volatile elements: operating system, policies;
  level: Virtual Infrastructure Level
- **Virtual components**
  volatile elements: hypervisor, computational infrastructure components (server, network and storage:);
  level: Virtual Infrastructure Level
- **Physical network infrastructure components**
  volatile elements: network components, firmware updates, additional hardware features;
  level: Physical Infrastructure Level

We revise the SECCRIT architecture based on aforementioned properties in the following subsections, in respect to the three dimensions defined in Section II and dynamic properties of individual component.

1) Component Assurance Properties: The referred SECCRIT architectural model enables the fine-grained specification of privacy, security and resilience requirements, which are

upheld by the cloud infrastructure. Such objects should also considered when talking about assurance. We hence to use this model in order to illustrate its individual components,(i.e. marked red on Figure 1). A property change could imply a new assurance level for the individual component and also the entire cloud service. Therefore, we abstract these individual components first per the following abstraction levels: Application Level, Virtual Infrastructure Level and Physical Infrastructure Level. Afterwards, each individual component should be independently assessed. The next step is to aggregate individual assessments per an abstraction level, and finally to aggregate abstraction level in a bottom-up approach for providing a holistic assurance assessment.

2) Dynamic Assurance Properties: Dynamic parameters of individual components,(i.e. the volatile objects mentioned above) cause the deviation of assurance during time despite the component, level, observed system or assurance element. For example, in case of dynamic allocation of additional virtual components (volatile objects), the aggregated assurance level of the service provided had to be re-evaluated automatically in time. The end goal is always to merge the assurance to assess an entire cloud service with an aggregated assurance. Another potential use cease where under the consideration of a case in which a self-healing mechanism adds a new component to the infrastructure layer to support recovery from an attack, it is not sure that the new component has the same assurance level like the other ones. Hence a low assurance level might not have such a heavy impact on the tenant system or the overall assurance of the service provided.

3) Dependability in component based systems: The mutual interaction of individual system components on various level is a mandatory prerequisite for delivering overall service. In order to address assurance in such system, the dependability is considered as an important property of assurance that should be included in the assessment.

C. Assurance Evaluation criteria

To properly address the research objectives due to the above mentioned objects we conduct the evaluation of the state-of-the-art related towards assurance in the Cloud to the best of our knowledge. Hence for qualitative assessment we provide the following set of evaluation criteria:

1) Assurance in the Cloud
2) Geo-locality
3) Homogeneous system
4) Heterogeneous system
5) Static infrastructure assessment
6) Dynamic infrastructure assessment
7) Data/Information assurance
8) System/Service assurance
9) Flexibility towards the evaluated system
10) Continuous assurance
11) Information assurance Definition
12) Aggregation of assurance

These criteria set is derived in according to the research objectives, the property set for assessing assurance and the SECCRIT architecture, respectively. The prior concern was to evaluate the existing work related with assurance for hosting Critical infrastructures in Cloud environments. Then, we considered the scalability of services in Cloud over difference geographical boundaries. Next point that we address is the system architecture depending on how the analysis was approached (holistic view as a single entity or Granular view - component based). The follow-up to the previous point distinguishes how the components in the system are considered (Static or dynamic) and if they address the two assurance elements (Information and system) addressed in the extension of our assurance definition. Clearly we wanted also to see how the proposed work is flexible towards the type of the system being evaluated. One of the points included from the research questions were the capability of the system to aggregate the assurance and provide continuous assurance, respectively. Finally we wanted to see who provided the formal definition of the assurance.

IV. STATE OF THE ART EVALUATION

We summarize, to the best of our knowledge, existing guidelines, methodologies, standards and approaches of closely related projects in respect to Assurance of Critical infrastructures hosted on top of the Cloud ecosystem. In particular, we investigate how the existing approaches confront the challenges and our research objectives mentioned in the previous section.

A. Guidelines

1) IT assurance Guide by COBIT: The goal of COBIT's IT Assurance Guideline [12] is to support and guide enterprises to leverage COBIT framework for variety of IT assurance activities. The guide is designed to support efficient and effective development of IT assurance initiatives, providing guidance on planning, scoping and executing assurance reviews using a road map based on well-accepted assurance approaches. The IT Assurance Guide provides assurance advice at the process and the control objective level. Furthermore, the guideline also implements the assessment processes in respect with the business plan, through the following three stages: planning, scoping and executing. First phase defines the universe of the assurance (the observed entities), selects an IT control framework, defines the set of preferred objectives, performs high level assessment and risk assurance planning. The second phase, defines in respect with the business model IT goals and key processes, resources and custom control objectives. The final phase refines the understanding of IT assurance subject and the scope of key control objectives, tests effectiveness and outcome of the key control objectives, setups the final conclusion and documents the impact on control weaknesses. COBIT guideline offers a fine grained analysis of the system with in respect to business goals, however it leaks the support for critical infrastructure and assurance in respect to Cloud ecosystems, locality issues, and aggregation of assurance.

2) Information Technology Assurance Framework: Information Technology Assurance Framework (ITAF) [13] is a comprehensive best practice guideline that provides design, guidance, implementation and reporting of IT audits and assurance assignments, defines concepts and terminologies in respect to IT assurance, and establishes set of reporting and auditing requirements. ITAF is composed of three standard guidelines: General set of standards, Performance Standards and Reporting Standards. The framework also operates and addresses other guidelines such as COBIT, ITIL, (ISO)/IEC 27000 standards, IT Control Objectives, IT Governance Domain Practices and Competencies, within the scope of his work to assess the IT infrastructure.

The framework is a adhering the above mentioned standards as a set of relevant requirements of an IT professional dealing with this IT assurance, and more tending towards guideline

for best practices for business and IT processes, in respect of assurance and audit standards. Therefore, making it a well structured, comprehensive and eligible best practice for IT business processes evaluation.
The ITAF derives best practices and strategic approaches to provide holistic assurance of a system, however doesn't address neither the critical or cloud infrastructures.

3) Cloud Computing Information Assurance Framework: ENISA's Information Assurance Framework [14] derives set of assurance criteria for: assessment of the risk for adopting cloud technologies, comparing various distinct cloud offerings, business and management process analysis and system policies. The framework is interesting only in terms of risk analysis for adopting cloud services, in our case this would be adopting critical infrastructure services, otherwise it cannot support more comprehensive analysis that we require.

4) National Security Agency Information Assurance Directorate: National Security Agency Information Assurance Directorate [15] provides an exhaustive assessment of the maturity and suitability of relevant IA technologies for meeting information assurance required capabilities. The directorate highlights four main cornerstones: Assured Information Sharing, Highly Available Enterprise, Assured Enterprise Management and Control and Cyber Situational Awareness and Network Defense. The cornerstones are mapped to Information Assurance System Enablers (Identification & Authentication, Policy Based Access Control, Protection of User Information, Dynamic Policy Management, Assured Resource Allocation, Network Defense & Situational Awareness and Management of IA Mechanisms & Assets) for a more convenient analysis and organization. The IA directorate advocates methodologies and best practices that should be conducted in order to achieve the assurance IA Components. Fine granulation is achieved through components and system enablers what are wrapped up with Information Assurance cornerstones. IA system enablers are mapped to sets of technology categories and mechanism, therefore regardless of the ability to wide and comprehensive application, the directorate is still repelling to changes.
Information Assurance Directorate addresses the problem of critical infrastructures in the scope of his work, but unfortunately without concerning the issues (e.g. locality issues which are also covered with our evaluation) relevant to hosting it on top of cloud infrastructures.

5) Handbook for Information Assurance Security Policy: This Handbook [16] is used to derive information assurance security policies complied with federal laws and regulations. The primary focus of this document are policies and guidelines that supports the IA Security Program in protecting the confidentiality, integrity, and availability of the Departments systems and information life cycle. Additionally, the handbook is reinforced through a series of standards, directives, and other procedures documents that address specific aspects of the IA Security Policy.
The handbook advocates set of management, operational and technical controls that undergo the referenced various guidelines and standards from the Office of Management and Budget, National Institute of Standards and Technology, General Services Administration and the Office of Personnel Management. Therefore it doesn't meet our objectives for supporting dynamic and flexible systems, continuous assurance, critical infrastructures in cloud environments or geo-locality issues in distributed environments.

6) Department of Defense Directives 8500.01 and 8500.02: Information assurance integrated in Department of Defense (DoD) Directives 8500.01 and 8500.02 [17], [18] derive a set of requirements that should be identified and included in the design, acquisition, installation, upgrade, or replacement of any information system within DoD. Whereby directive is pointer towards maintaining an appropriate level of confidentiality, integrity, authentication, non-repudiation, and availability. Directive efficiently utilizes defense-in-depth approach that integrates the capabilities of personnel, operations, and technology.
Both directives were built upon DoD's ICT systems, and therefore address information assurance concerns that are only related to DoD's systems, which makes them less applicable and limiting for broader usage.

7) Deputy Assistant Secretary of Defense for Cyber, Identity, and Information Assurance Strategy : This document [5] derives strategies to organize for unity of purpose and speed of actions, enable secure mission-driven access to information and services, anticipate and prevent successful attacks on data and networks, and prepare for and operate through cyber degradation or attack. The focus of this work is to establish a narrow-down set of strategic activities for maintaining and insuring information assurance, which unfortunately covers only a minor part of our objectives regarding assurance.

8) Information Assurance Governance Framework: The Information Assurance Governance Framework [19] is focused to derive functional and managerial hierarchy for information assurance, risk management procedures and guidelines, and to identify mechanisms, procedures and best practices for facilitating information assurance. The main focus of the framework is pointed on facilitating management and risk confidence of the stakeholders. Therefore, this framework is oriented on business aspects rather then technical which we are addressing as the priority of our work.

9) Common Criteria: The Common Criteria for Information Technology Security Evaluation framework[2] [20] is a well-known approach to apply rigorous engineering methods and processes to the design and development of security and critical IT systems. Common Criteria (CC) provide the process of specification, implementation, and evaluation of security-critical, high-assurance systems in a rigorous and standardized manner. The key concept of CC is that by testing a security product against defined security properties of the product, it can be determined with high confidence if the product can actually meet its claims. In a CC evaluation process, a Target Of Evaluation (TOE) is the product or system under evaluation. A user or a user community identifies common security requirements on a class of devices or systems such as access control devices and systems or key management systems in the Protection Profile (PP) document. A Security Target (ST) document contains the IT security requirements of the TOE and specifies the functional and assurance measures offered by the TOE to meet these requirements. The effort of the evaluation process is ranked numerically from one to seven in Evaluation Assurance Levels (EAL). CC provides not only a benchmark for security "'due diligence"' checking, but also assurance on the design, development, deployment, and life-cycle handling of security-critical systems. CC can significantly increase the security of a software/hardware system

[2]Common Criteria, http://www.commoncriteriaportal.org/

as well as the confidence of the end-user of the system by emphasizing good and comprehensive documentation during the system design and development phase. At this the system development team has security as its main objective from the very beginning. There is also a raise of awareness related with security problems throughout the system's design and development phases.
Regardless of the rich set of features facilitated by the framework, it still doesn't support the aggregation of different assurance levels for individual components, concerns the systems hosted in Cloud, or derive a continuous assurance. Therefore this has to be resolved in order to overcome the problems mentioned in the introduction section of this paper. However, the approach of Common Criteria offers a solid foundation for building components based assurance framework for critical infrastructures in the Cloud ecosystem.

10)Cloud Trust Protocol: The Cloud Trust Protocol (CTP) is the mechanism which offers cloud users to request and acquire information about the elements of transparency as applied to cloud service providers. The primary purpose of the CTP and the elements of transparency is to generate evidence-based confidence that everything that is claimed to be happening in the cloud is indeed happening as described. This is a classic application of the definition of digital trust. And, assured of such evidence, cloud consumers become liberated to bring more sensitive and valuable business functions to the cloud, and reap even larger pay-offs. With the CTP cloud consumers are provided a way to find out important pieces of information concerning the compliance, security, privacy, integrity, and operational security history of service elements being performed "in the cloud".
These important pieces of information are known as the "elements of transparency", and they deliver evidence about essential security configuration and operational characteristics for systems deployed in the cloud. The elements of transparency empower the cloud consumer with the right information to make the right choices about what processing and data to put in the cloud or leave in the cloud, and to decide which cloud is best suited to satisfy processing needs. This is the nature of digital trust, and reinforces again why such reclaimed transparency is so essential to new enterprise value creation. Information transparency is at the root of digital trust, and thus the source of value capture and pay-off. [21]
Cloud Trust Protocol facilitates data acquisition over distinct cloud providers is a large benefit towards achieving transparency but unfortunately it doesn't assurance of the actions been really conducted from the provider (for example location of the data, how can we know that some part or the whole data set hasn't been replicated on some other location).

B. Projects

1) Cumulus: CUMULUS is aligned with the recommendations of a recent industrial consultation to the European Commission which identified cloud certification as an enabling technology for building trust for end users through the deployment of standards and certification schemes relevant to cloud solutions, and included it in the ten key recommendations and actions for a cloud strategy in Europe [22].The project develops an integrated framework of models, processes and tools supporting the certification of security properties of infrastructure (IaaS), platform (PaaS) and software application layer (SaaS) services in cloud. The framework will bring service users, service providers and cloud suppliers to work together with certification authorities in order to ensure security certificate validity in the ever-changing cloud environment. The project relies on multiple types of evidence regarding security, including service testing and monitoring data and trusted computing proofs, and based on models for hybrid, incremental and multi-layer security certification. To ensure large-scale industrial applicability, this framework will be evaluated in reference to cloud application scenarios in some key industrial domains, namely smart cities and eHealth services and applications. Therefore, the certification model is an attractive solution for handling security parameters that have to be meet inside of a system. However, at the moment the approach addresses only single level certification within his scope without the aggregation of the levels, but it addresses the same core problem of meeting security requirements.

2) A4Cloud: The Cloud Accountability Project (or A4Cloud for short) focuses on the accountability for cloud and other future internet services as the most critical prerequisite for effective governance and control of corporate and private data processed by cloud-based IT services. The research being conducted in the project will increase trust in cloud computing by devising methods and tools, through which cloud stakeholders can be made accountable for the privacy and confidentiality of information held in the cloud. These methods and tools will combine risk analysis, policy enforcement, monitoring and compliance auditing. They will contribute to the governance of cloud activities, providing transparency and assisting legal, regulatory and socio-economic policy enforcement. [23]In [24], [25], as a part of the A4Cloud project, authors comprehensively address accountability pointed towards governance.
A4Cloud project addresses assurance indirectly under the scope of accountability within respect to the data governance. The comprehensive approach conducted to ensure the accountability correlates with our scope and goals, the difference that we base our work on hosting critical infrastructures on top of cloud stack.

3) MYSEA: The Monterey Security Architecture [26](MYSEA)[3] is a research project to build a robust enterprise-level architecture that provides multi-domain authentication and security policy enforcement. The MYSEA cloud consists of high-assurance servers and authentication components for security services. The high assurance of MYSEA cloud is built on a trusted server (i.e., an EAL5-augmented trusted platform) and authentication component (i.e., an EAL7 Least Privilege Separation Kernel). Originally aiming at composing secure distributed systems using commercial off-the-shelf components, some of the results from the MYSEA project might also be applicable to cloud computing environment.
Regarding the topic of our paper, MYSEA only consists of a few components evaluated with a certain assurance level (trusted server and authentication component). There is no necessity of aggregating different assurance levels of different components. An advantage of this architecture is, that clients, respectively cloud service users, also are considered due to security reasons. In the case of a given assurance level framework, there is the gap of what is the right treatment of

[3]Monterey Security Architecture (MYSEA), Centres for Information Systems Security Studies and Research at Naval Postgraduate School, U.S., http://www.cisr.us/projects/mysea.html

an unprotected cloud service user which wants to connect to the service.

C. Discussion

We carried out a comprehensive state-of-the-art evaluation approaches, methodologies, procedures, guidelines and projects related with system and information assurance of critical infrastructures hosted on top of Cloud ecosystems. The Cloud ecosystems, as anticipated, can offer full support to internet scale critical applications (e.g. hospital systems and smart grid systems). Unfortunately, organizations refuse to outsource their resources, regardless if critical or not, without confidence that a proper set of actions and measures are undertaken to provide information and system assurance. The approaches such as mentioned in the work [27] support scalable critical applications over the Cloud infrastructure, by providing assurance to cloud users related with the trustworthiness of service delivery in a cloud environments, known as operational trust. Particular focus is on analyzing the most important properties (adaptability, scalability, resilience, availability and reliability) within a cloud, which enable assessments of the operational trustworthiness or effectiveness of a cloud provider for delivering these services. The assessment of operational trust enables cloud service users, auditors, collaborating cloud providers, and others to improve the decision making and quantifying cloud providers. Additionally in [28] authors advise a trust-overlay network over multiple data centers to implement a reputation system for establishing trust between service providers and data owners. In order to offer additional layer of security and trustworthiness data coloring and software watermarking techniques protect shared data objects and massively distributed software modules. These techniques safeguard multi-way authentication, enable single sign-on in the cloud, and tighten access control for sensitive data in both public and private clouds.

The outcome of our comprehensive state-of-the-art evaluation is presented in Table I that can be found on page 7. We classify two venues for our inquisition: Frameworks/Guidelines/Standards/Policies and assurance related Projects.

Primarily, we focused on inquiring and evaluating the work that covers the domain issues related with Critical infrastructures and Assurance. Although critical infrastructures are a specific and broad domain, additionally hosting them on top of Cloud infrastructure extends their perimeter and improves the performance, However also it raises new challenges related to security, privacy, availability, verifiability, etc. that we observer under the term of assurance. The National Security Agency (NSA) in the Information Assurance Directorate [15] and Department of Education (DoE) in the Handbook for Information Assurance Security Policy [16] reference in scope of their work general assurance requirements related with critical infrastructures. One of our main points of interest also related with the work done as a part of SECCRIT project is to investigate locality challenges and issues, therefore we include geo-locality concerns (e.g. legislative issues confronted by cross administrative and regional migrations). Geo-locality has been addressed by several large organizations, such as National Security Agency, ENISA, Department of Defense, Department of Education, and A4Cloud research project. The [14], [16], [18], [20], [21], [23], [26] referred to the geo-locality as as an obligatory part of a federal or local law, whereby in our case we would like to consider it as cross domain (geographical, federal, regional, administrative) issue required for assessing overall assurance. Next to our interest is the observation perspective of a system, where we wanted to investigate if the system was observed from a holistic or a homogeneous perspective. Majority of the work that was evaluated [12], [13], [14], [15], [16], [17], [18], [5], [26], [23], [22], [20], [26] derived their work in a holistic manner, with minority of approaches [14], [20], [23], [22], [26] that focused to observe system in a heterogeneous manner. Furthermore we wanted to see how does a particular state-of-the-art work observe the properties of a system over time. Therefore, we focused to evaluate if state-of-the-art work is capable of confronting dynamic system changes such as component, class, modules, vendor, etc. that can change their functionalities and characteristics. In particular, the work of [12], [13], [14], [15], [16], [5], [20], [23], [22] refers to a static system observation, whereby the work of [20], [12], [23], [22] due to their flexibility in the approaches are able to granulate system through components and deal with dynamic changes of a system. The next point of our evaluation is observed within respect to the definition of assurance and it's elements (data and service assurance). The majority of the evaluated state-of-the-art-work [12], [14], [15], [16], [17], [18], [5], [19], [20], [23], [26] derived the work in both system and information assurance, whereby the remaining work [13], [21], [22] didn't address this issue at all. Despite the fact that CUMULUS [22] doesn't directly address the assurance, the major benefit of their approach is the ability to continuously deliver assurance through the certificates that they deliver only per individual level. Furthermore we wanted to see who defines the assurance to avoid ambiguity of term being used in general manner. Unfortunately, only minor part of the evaluated work [13], [17], [18], [5], [19], [20], [23] formalized the assurance in form of a definition within respect of particular objectives. As the last point of our evaluation we inquire the capability of the state-of-the-art approaches to aggregate the assurance of individual components for evaluating a system as a whole. Only the approaches [20], [21], [23], [22] have addressed the problem of information aggregation to holistically observe some system.

V. CONCLUSION AND FUTURE WORK

We have identified a number of issues regarding assurance of CI in the cloud and identified short-comes via a comprehensive evaluation of existing approaches. As a result we propose as a part of the conclusion a new assurance approach and framework as the basis of our future work.

A. Conclusion

This work identified the set of problems, stated as research questions (Section 1),which address assurance for those systems that require specific care when hosting in Cloud environments (i.e. Critical infrastructures). Furthermore, we investigated the shortcomings of existing methodologies, guidelines, frameworks, standards and projects for supporting high assurance in Cloud environments and SECCRIT architecture, respectively. Our evaluation outcomes that the current work in Cloud environments lacks clarity and executability for identifying security requirements and security properties of higher-assurance systems for critical infrastructures in cloud computing. Considering our research objectives, the main drawback of the methodologies, guidelines, frameworks and standards for assessing assurance in cloud is the support for

TABLE I. EVALUATION OF THE STATE OF THE ART APPROACHES FOR ADDRESSING ASSURANCE IN CLOUD ECOSYSTEMS

	Frameworks/Guidelines/Standards										Projects		
	IT Assurance Guide	Information Technology Assurance Framework	Cloud Computing Information Assurance Framework	Information Assurance Directorate	Handbook for Information Assurance Security Policy	Directives 8500.01 and 8500.02	Deputy Assistant Secretary of Defense for Cyber, Identity, and Information Assurance Strategy	Information Assurance Governance Framework	Common Criteria for Information Technology Security Evaluation Framework	Cloud Trust Protocol	Certification infrastructure for multi-layer cloud services project	A4Cloud project	The Monterey Security Architecture
Assurance in the Cloud				✓	✓								✓
Geo-locality			✓		✓	✓			✓	✓		✓	
Homogen system	✓	✓	✓	✓	✓	✓	✓		✓		✓	✓	✓
Heterogeneous system			✓			✓			✓		✓	✓	✓
Static infrastructure assessment	✓	✓	✓	✓	✓		✓		✓		✓	✓	✓
Dynamic infrastructure assessment	✓								✓		✓	✓	
Data/Information assurance	✓		✓	✓	✓	✓	✓	✓	✓			✓	✓
System/Service assurance	✓		✓	✓	✓	✓	✓	✓	✓			✓	✓
Continuous assurance											✓		
Information assurance Definition		✓				✓	✓	✓	✓			✓	
Aggregation of assurance									✓	✓	✓	✓	

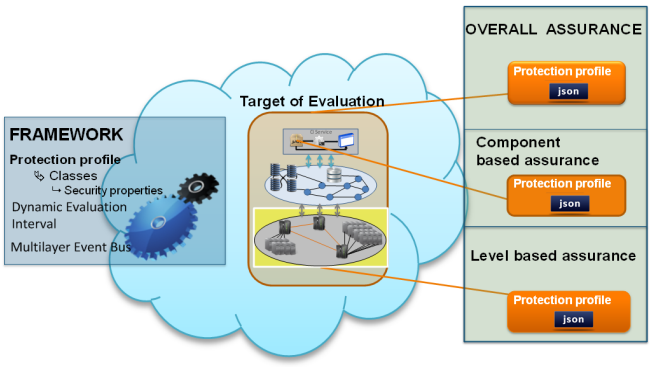

Fig. 2. Our Assurance Assessment Framework for assessing CI services hosted in the Cloud. The framework is based on classes (i.e. confidentiality, availability, etc.) which are motivated by Common Criteria [20] protection profiles. Protection profiles will be derived for the whole system, for each abstraction level and individual component. This is based on existing work [6], [29]

hosting CI in the Cloud. The evaluation showed that only the concepts used in the work of Common Criteria, CUMULUS and A4Cloud are partially eligible to resolve the challenges for hosting CI in the cloud. Whereby, Common Criteria allows us to evaluate traditional IT component based system, dependencies of components, comparability between the results of independent security evaluations and overlaps with the security consideration of our assurance definition. However, additional extensions of the framework are required to completely support our research objectives like aggregation of assurance, continuous assurance or automated assurance within respect Cloud ecosystem. CUMULUS project delivers the important feature certification of continuous monitoring [30] that we can adopt to support continuous assurance. A4Cloud addresses the issue of assurance in the cloud under the term of accountability and corporate data governance, which also doesn't completely fulfill our requirements.

Despite that there are approaches derived for addressing assurance, mostly addressing information assurance, the evaluation showed that a framework for extensive analysis of assurance in Cloud ecosystem is required. Therefore, an independent framework for addressing assurance in Cloud-based systems would require to address the following: assurance of the systems hosted on top of the Cloud, delivering assurance continuously at any point of time; classifying assurance per abstraction levels and components, based on the propose architecture; technology independent assessment; aggregating of assurance in automated manner.

Motivated by the outcome of our evaluation which clearly outlined the shortcomings of existing approaches for supporting the assurance in Cloud ecosystems and lack of any kind of solutions that would support it, we propose an independent *Assurance Assessment Framework* for assessing Critical infrastructures hosted on the top of the Cloud ecosystems, Figure 2. The proposed Assurance Assessment Framework, founded on our extended assurance definition, distinguishes the assurance in the system prior to the assurance elements (system assurance or information assurance). Each individual assurance element is additionally classified per component, layer or component dependency. For the purpose of the SECCRIT project we outlined dynamic properties per abstraction levels (user, application, virtual and physical infrastructure). Our Framework defines the Protection profile [20] in respect with dynamic properties of a component, layer or dependency, dynamic assessment interval [6] and Multilayer Event Bus [29]. Protection profile is composed of Classes (i.e. availability, confidentiality, integrity, etc.) where each individual class is depicted by a set of security properties. This framework configuration allows us to deliver customized assurance assessment per individual

component that can aggregated the assurance per abstraction levels, and finally overall assurance.

B. Future Work

The Assurance Assessment Framework was founded on the work delivered within the scope of SECCRIT project deliverables (D2.1[4],D2.2[5], D3.1[6] and D5.1[7]), that derived requirements of the use case scenarios, vulnerability catalogue, APIs for information acquiring, and auditing processes. Our future goal is to build our Assessment Framework for delivering continuous assurance by extend the well-known concepts of Common Criteria class based approach [20] to aggregate assurance in continuous manner, and the concepts of CUMULUS certification of continuous monitoring module [30].
For the empirical evaluation we will focus to build a proof of concept for acquiring information/evidence based on work [29], [7], [8], [9], per abstraction levels. To overcome the fallback of restricted information acquisition per level in case of different stakeholders and consider the work of [31], [32], [33], [34], [35] regarding privacy and security related concerns, as an alternative we will integrate and rely on the services offered by the Cloud provider (i.e. SLA, monitoring services or trust protocols).

ACKNOWLEDGMENT

This work has been funded from the European Unions Seventh Framework Programme for research as the SECCRIT project under grant agreement no 312758.

REFERENCES

[1] C. C. John Moteff, S. John Fischer Resources, and I. Division, "Critical infrastructures: What makes an infrastructure critical?" Congressional Research Service The Library of Congress 101 Independence Ave. SE Washington, DC 20540, Report for Congress, 2003.

[2] J. Moteff, C. Copeland, and J. Fischer, "Critical infrastructures: What makes an infrastructure critical?" DTIC Document, 2003.

[3] W. Krger, "Critical infrastructures at risk: A need for a new conceptual approach and extended analytical tools," *Reliability Engineering & System Safety*, vol. 93, no. 12, p. 17811787, 2008.

[4] L. Wu and R. Buyya, "Service level agreement (sla) in utility computing systems," *CoRR*, vol. abs/1010.2881, 2010.

[5] D. of Defense, *Deputy Assistant Secretary of Defense for Cyber, Identity, and Information Assurance Strategy*, 1st ed., Department of Defense, August 2009. [Online]. Available: http://dodcio.defense.gov

[6] M. Tauber, G. N. C. Kirby, and A. Dearle, "Autonomic management of maintenance scheduling in chord," *CoRR*, vol. abs/1006.1578, 2010.

[7] K. Mahbub, G. Spanoudakis, and T. Tsigkritis, "Translation of slas into monitoring specifications," in *Service Level Agreements for Cloud Computing*, P. Wieder, J. M. Butler, W. Theilmann, and R. Yahyapour, Eds. Springer New York, 2011.

[8] G. Spanoudakis, C. Kloukinas, and K. Mahbub, "The serenity runtime monitoring framework," in *Security and Dependability for Ambient Intelligence*, ser. Advances in Information Security, S. Kokolakis, A. M. Gmez, and G. Spanoudakis, Eds. Springer US, 2009, vol. 45.

[9] H. Foster and G. Spanoudakis, "Advanced service monitoring configurations with sla decomposition and selection," in *Proceedings of the 2011 ACM Symposium on Applied Computing*, ser. SAC '11. New York, NY, USA: ACM, 2011.

[4]Deliverable 2.1 - Report on requirements and use cases, https://www.seccrit.eu/upload/D2-1-Report_on_requirements_and_use_cases-v2.0

[5]Deliverable 2.2 - Legal fundamentals, https://www.seccrit.eu/upload/D2-2-Legal-Guidance-v2.0

[6]Deliverable 3.1 - Methodology for Risk Assessment and Management, https://www.seccrit.eu/upload/D3-1-Methodology-for-Risk-Assessment-and-Management

[7]Deliverable 5.1 - Design and API for Audit Trails and Root-Cause Analysis, https://www.seccrit.eu/upload/D5.1-Audit-Trails

[10] F. P. J. H. P. S. Marcus Schoeller, Roland Bless, "An architectural model for deploying critical infrastructure services in the cloud," *IEEE CloudCom 2013*, 2013.

[11] R. Bless, M. Flittner, J. Horneber, D. Hutchison, C. Jung, F. Pallas, M. Schller, S. N. H. ul Shirazi, S. Simpson, and P. Smith, "Whitepaper "af 1.0" seccrit architectural framework," 2014.

[12] COBIT, *IT Assurance Guide: Using COBIT*, Control Objectives for Information and related Technology, 2007, information Systems Audit and Control Association.

[13] ITAF, *Information Technology Assurance Framework*, 2nd ed., Information Systems Audit and Control Association, 2013.

[14] ENISA, *Cloud Computing Information Assurance Framework*, 1st ed., European Union Agency for Network and Information Security, 2009. [Online]. Available: http://www.enisa.europa.eu/

[15] NSA, *Information Assurance Directorate*, 1st ed., National Security Agency, 2010.

[16] D. of Education, *Handbook for Information Assurance Security Policy*, 1st ed., U.S. Department of Educations, 2005. [Online]. Available: http://www2.ed.gov/fund/contract/about/acs/acshbocio01.doc

[17] DoD, *Directives 8500.01*, 2nd ed., Department of Defense, May 2005. [Online]. Available: www.prim.osd.mil/Documents

[18] ——, *Directives 8500.02*, 1st ed., Department of Defense, 2003. [Online]. Available: www.prim.osd.mil/Documents

[19] CSIA, *Information Assurance Governance Framework*, 1st ed., CSIA, 2004. [Online]. Available: www.cabinetoffice.gov.uk

[20] CC, *Common Criteria for Information Technology Security Evaluation*, CCDB USB Working Group, 2012, part 1-3. [Online]. Available: http://www.commoncriteriaportal.org

[21] CSA, "Introduction to cloudtrust protocol," 2011. [Online]. Available: https://cloudsecurityalliance.org/research/ctp/

[22] CUMULUS, "Certification infrastructure for multi-layer cloud services project (cumulus)," 2012. [Online]. Available: http://www.cumulus-project.eu/index.php/description

[23] A4Cloud, "Cloud accountability project (a4cloud)," 2013. [Online]. Available: http://www.a4cloud.eu/

[24] M. Felici, T. Koulouris, and S. Pearson, "Accountability for data governance in cloud ecosystems."

[25] S. Pearson, "Toward accountability in the cloud." *IEEE Internet Computing*, vol. 15, no. 4, 2011.

[26] C. E. Irvine, T. D. Nguyen, D. J. Shifflett, T. E. Levin, J. Khosalim, C. Prince, P. C. Clark, and M. Gondree, "Mysea: The monterey security architecture," p. 3948, 2009.

[27] I. M. Abbadi, "Operational Trust in Cloud's Environment," *IEEE Symposium on Computers and Communications,*, p. 141145, 2011.

[28] D. L. Kai Hwang, "Trusted Cloud Computing with Secure Resources and Data Coloring," *IEEE Internet Computing*, vol. 14, Sept. 2010.

[29] M. Florian, S. Paudel, and M. Tauber, "Trustworthy evidence gathering mechanism for multilayer cloud compliance," in *ICITST*, 2013.

[30] M. Krotsiani, G. Spanoudakis, and K. Mahbub, "Incremental certification of cloud services," in *SECURWARE 2013, The Seventh International Conference on Emerging Security Information, Systems and Technologies*, 2013, p. 7280.

[31] Z. Fadlullah, T. Taleb, A. Vasilakos, M. Guizani, and N. Kato, "DTRAB: Combating Against Attacks on Encrypted Protocols Through Traffic-Feature Analysis," *IEEE/ACM Transactions on Networking*, 2010.

[32] L. Wei, H. Zhu, Z. Cao, W. Jia, and A. Vasilakos, "SecCloud: Bridging Secure Storage and Computation in Cloud," in *Distributed Computing Systems Workshops (ICDCSW)*, June 2010, p. 5261.

[33] L. Wei, H. Zhu, Z. Cao, X. Dong, W. Jia, Y. Chen, and A. V. Vasilakos, "Security and Privacy for Storage and Computation in Cloud Computing," *Inf. Sci.*, vol. 258, p. 371386, Feb. 2014.

[34] B. Liu, J. Bi, and A. Vasilakos, "Toward incentivizing anti-spoofing deployment," *Information Forensics and Security, IEEE Transactions on*, vol. 9, no. 3, p. 436450, March 2014.

[35] N. Fernando, S. W. Loke, and W. Rahayu, "Mobile Cloud Computing: A Survey," *Future Gener. Comput. Syst.*, vol. 29, 2013.

Impact of Critical Infrastructure Requirements on Service Migration Guidelines to the Cloud

The paper "Impact of Critical Infrastructure Requirements on Service Migration Guidelines to the Cloud" was published on IEEE International Conference on Future Internet of Things and Cloud 2015 in Rome.

The paper can be found online under the following link: `http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7300790&newsearch=true&queryText=mpact%20of%20Critical%20Infrastructure%20Requirements%20on%20Service%20Migration%20Guidelines%20to%20the%20Cloud` ublished on IEEE

Impact of Critical Infrastructure Requirements on Service Migration Guidelines to the Cloud

Christian Wagner, Aleksandar Hudic,
Silia Maksuti, Markus Tauber
Austrian Institute of Technology
Vienna, Austria
{christian.wagner, aleksandar.hudic,
maksuti.silia.fl@ait.ac.at, markus.tauber}@ait.ac.at

Frank Pallas
Karlsruhe Institute of Technology
Karlsruhe, Germany
frank.pallas@kit.edu

***Abstract*—A high level of information security in critical infrastructure IT systems and services has to be preserved when migrating their IT services to the cloud. Often various legislative and security constraints have to be met in line with best practice guidelines and international standards to perform the migration. To support the critical infrastructure providers in migrating their services to the cloud we are developing a process based migration guideline for critical infrastructure providers focusing on information security. First of all we investigate, via questionnaires, how the importance of individual security topics covered in such guidelines differentiates between industry stakeholders and critical infrastructure providers. This supports the selection of relevant security topics and the considered guidelines and standards, which we survey in search for common relevant security topics. Subsequently we present the analysis of the above-mentioned security requirements and how they affect a here developed taxonomy for a process-based security guideline. Furthermore we present potential service migration use cases and how our methodology would affect the migration of secure critical infrastructure services.**

***Index Terms*—survey analysis; security requirements; critical infrastructure; industry; security guideline, cloud migration**

I. INTRODUCTION

Enterprises recognized the cloud paradigm as an opportunistic business strategy to remain competitiveness, meeting business objectives, increasing performance and reducing costs [1], [2]. The utilization of services across a layered distributed architecture, that is the very nature of cloud computing [3], offers tremendous advantages over a traditional computing paradigm [4]. Cost reduction is one of the main benefits which affect both cloud provider and cloud customer [4]. Therefore, migration of services from expensive enterprise IT infrastructures to the cloud became a prominent and cost-efficient solution [5], [6], [7]. Although, the migration into the cloud offers various benefits [8], [9] primarily in terms of finances, often it is the case that services that are intended to be ”cloudified” (i.e. migrated to cloud) are not designed for distributed computing. Thus, additional steps that include detailed analysis and setting up guidelines for migrating services are required [10], [11], [12].

Some of the proclaimed benefits make the cloud also attractive to organizations with high protection requirements, such as critical infrastructures (e.g. telecommunication organizations, the electric power industry, healthcare services, or agriculture companies). However, when considering such a scenario for critical infrastructure services, the potential consequences of a malfunction are of major significance, leading to such systems and services typically being subject to strict regulations in matters of security. Therefore, appropriate measures for maintaining and accomplishing intended information security levels are required from a critical infrastructure providers' perspective. IT systems and services used for managing critical infrastructures require a large amount of resources, and hence critical infrastructure providers often host their own infrastructure or may join resources with other similar organizations. In any case multi-tenant and multi-layer issues apply in these scenarios in a similar way as for common IT businesses.

978-1-4673-8103-1/15/$31.00 © 2015 IEEE

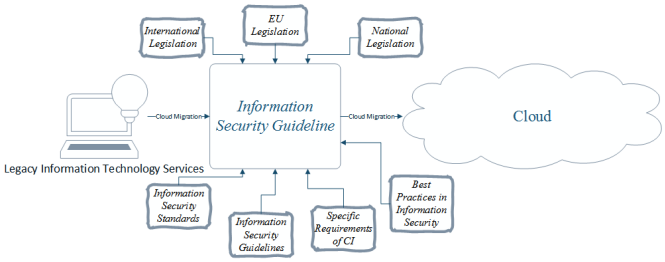

Fig. 1. Information Security Guideline Overview - Topics which determine a cloudification process

In this paper we present our approach in constructing an information security guideline, in form of a life cycle, for the cloud migration phase of critical infrastructure IT services (Figure 1). In our research we have first investigated the differences in security requirements between industry, and critical infrastructure providers by means of questionnaires. With the obtained information from the analysis of the returned questionnaires as well as an extensive literature analysis we develop a taxonomy which we set in relation to relevant best practice guidelines. Hence our main contributions are threefold:

- Analysis of differences in information security requirements of critical infrastructure providers and industrial stakeholders with respect to cloud computing.
- Survey of standards and guideline topics, related to the identified requirements and a taxonomy in order to

represent the results.
- An approach for a process-based information security guideline, including - in the survey- identified process-steps, for the migration of critical infrastructure services to the cloud.

The remainder of this paper is as follows: We evaluate related work in section II. In section III the survey methods and the results of an information security requirements survey among industry and critical infrastructure providers are presented. In section IV the survey results are applied to a taxonomy imbedded in a process-based cloudification life cycle. We present a conclusion and outline of future work in section V.

II. RELATED WORK

In this section we discuss the recent security challenges, methodologies, guidelines and standards with respect to cloud services with high security requirements.

A. Critical infrastructure Security challenges

Despite the attractive economical and performance benefits lack of security (e.g. lack of transparency, data privacy, trust, data lock-In, data loss) still remains the main obstacle for migrating services in to the cloud. This is especially emphasized when taking in to the account services with high security requirements such as critical infrastructure services. Hence, the analysis of security issues in cloud attracted broad research interest [13], [14], [15], [16], [17], [18], [19], [20], [21], [13]. However, majority of the research community concluded that the current security methods and techniques in cloud are not mature enough to reliably support hosting services in cloud. Nevertheless, solutions for migrating enterprise services to the cloud are emerging constantly [10], [11], [6], [12], [5], [7].

Younis et. al. [21] based on their detailed security analysis for various critical infrastructure providers, outlined major security issues in the cloud that hinder the migration of critical infrastructure services. Alcaraz and Zeadally in their survey [13] highlighted the vital role of critical infrastructures in modern society. However, they elaborated security challenges of critical infrastructure assets are mainly focusing on the industrial control systems (SCADA). In addition, authors evaluate the compliance of critical control systems towards standards, recommendations and guidelines.

B. Migration Concepts and Methodologies

Khajeh et. al elaborated in their work [11], [6] the migration of enterprise IT services to the cloud in context of financial and socio-technical enterprise issues which should be considered during migration. In addition, the authors elaborate the decision making process for service migration with two following tools: cost modeling, and benefits and risk assessment. Kaisler and Money investigated in their work [10] the compatibility of the service migration approach with the cloud computing paradigm by addressing various issues(acquisition, implementation, security, usage reporting, valuation and legislative). Fehling et. al. advocate in their work [12] best practices for addressing service migration challenges in context of migration patterns demonstrated on a web based application. Sun and Li perform effort estimation on infrastructure level by using tool for that automatically migrates configuration of workload from physical platform to visualized platform. in their systematic literature review Jamshidi et. al. [7] identify and systemically analyze existing research on legacy software migration to the cloud. The outcome of the research identified the importance of a comprehensive migration framework, which would taxonomically classify and compare various studies for cloud service migration. Paudel et. al. [22] analyzed how mitigation options for identified open security issues for critical infrastructures in the cloud point to individual aspects of standards and guidelines.

C. Standards and guidelines

ENISA generalizes security issues of cloud computing from Critical Information Infrastructure Protection (CIIP) perspective [23]. Additionally, authors discuss the risk assessment and security measures related with CIIP. Although our work is closely related with the work from ENISA, we are more focused on outlining open issues of hosting critical infrastructure services in the cloud.

The National Institute of Standards and Technology (NIST) published the framework for improving Critical Infrastructure Cybersecurity[24]. The framework provides a set of guidelines for developing individual organizational profiles, by aligning cybersecurity activities with business requirements, risk tolerances and resources.

We have identified in the related work that there is some existing work which addresses the service migration methodologies and processes in a generic way. There is also a part of the research community which elaborate on security issues and requirements referred to cloud computing. Furthermore, there are international standards and guidelines available to deal with the protection of critical infrastructure providers. However, there is no uniform solution that addresses the above mentioned challenges, critical infrastructure protection, and secure service migration to cloud environments.

III. INFORMATION SECURITY REQUIREMENTS ANALYSIS

A. Research methodology

To highlight and analyze the differences between industry and critical infrastructure providers information security requirements we performed an extensive survey among industry and academic experts. Thus, we distributed the questionnaire at various events with a cloud computing focus. Furthermore, to acquire the results from broader audience of professionals we also offered an online version of our questionnaire. Finally, we acquired 111 participants (72 via events and 39 online). Answers from academia, where listed, are only used as control sample.

1) Normalization of the results: For most of the questions in the questionnaire, survey participants could rank their opinion according to their importance (i.e. not at all important, slightly important, important, fairly important, very important, and no opinion).

For the analysis of the survey we chose the following normalization formula:

$$\frac{actualrepliesperanswer option}{samplesizeperdomain} \times 100 \times weight \quad (1)$$

In addition the weight values, shown in Table I, were used calculating the normalized output in the above mentioned equation.

TABLE I
WEIGHTING SCALE

Answer Option	Weight
No Opinion	$\log_{10}$ 1
Not at all important	$\log_{10}$ 1
Slightly important	$\log_{10}$ 2
Important	$\log_{10}$ 3
Fairly important	$\log_{10}$ 4
Very important	$\log_{10}$ 5

In the nomenclature of the possible answers the results presented in this survey analysis therefore have the following meaning:

TABLE II
MEANING OF NORMALIZED IMPORTANCE

Range	Meaning
0 % - 43 %	Slightly important
44 % - 68 %	Important
69 % - 86 %	Fairly important
87 % - 100 %	Very important

B. Evaluation of survey results

In the analysis of the provided questionnaire, we show the importance of the NIST cloud characteristics for the industry and critical infrastructure providers and their security needs for these characteristics. We in particular consider the aspect of the geolocation of cloud providers. Furthermore we indicate that the importance of information security for the cloudification of various exemplary IT service for the respective domains. Based on a pre-selected list of security controls, we analyze their importance for the industry and critical infrastructure providers. This is a starting point for the creation of the taxonomy for the process-based information security guideline (chapter IV).

The results of this survey analysis address the following questions:

- Which typical information security requirements (availability, integrity, confidentiality, auditing) are most relevant for critical infrastructure providers for applying cloud computing business models to their IT services?
- Which security controls related to cloud computing environments do critical infrastructure providers consider as important for IT service cloudification?
- How do the findings of this survey analysis influence the taxonomy for a cloud migration guideline for critical infrastructure providers?

Within the following six paragraphs we summarize and justify the most relevant outputs of our survey.

1) Company affiliation of survey respondents: In order for being able to make differentiated statements the survey participants were asked to specify their company affiliation. Out of all 111 respondents

- 31 individuals (28 %) have stated to be affiliated to organization type academia,
- 46 (41 %) to industry,
- 23 individuals (21 %) to critical infrastructure provider, and
- 11 (10 %) to another, undefined organization type.

2) Importance of the geolocation of the cloud provider and relevance of individual cloud computing characteristics: In the survey, besides the elicitation of security requirements of industry and critical infrastructure providers, the respondents were asked some general questions about 1) the importance of the NIST cloud computing characteristics (on-demand self-service, broad network access, resource pooling, rapid elasticity, measured service) , and 2) the importance of the geolocation of the cloud provider.

With respect to economic espionage, the location of the cloud provider and the data center is very often proclaimed as an important topic [25]. Furthermore, location is also highly relevant from the legal/regulatory perspective, particularly including European data protection law. Our analysis (Figure 2) shows that geolocation is in fact an important element for critical infrastructure providers when selecting cloud providers (12 % higher compared to the industry domain). The total values are: 78 % importance for critical infrastructure domain, 66 % for industry, and 67 % for academia (the control sample).

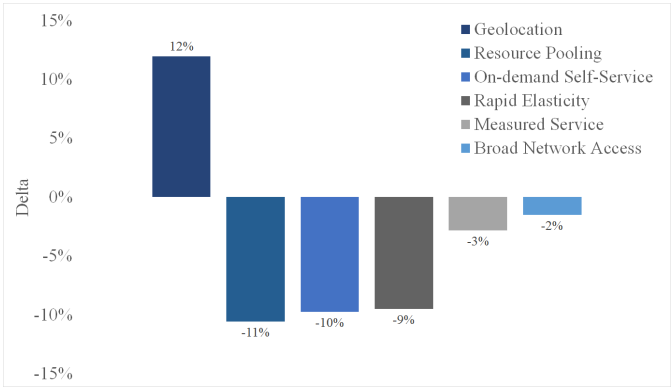

Fig. 2. Values show that regarding the here presented comparison of interest in NIST cloud characteristics of critical infrastructure providers with industry stakeholders, only geolocation is significantly more important for critical infrastructure providers (based on our importance metric).

With respect to the NIST cloud computing characteristics the delta values (Figure 2) show that broad network access is the most important cloud characteristic for critical infrastructure providers, whereas the resource pooling possibilities cloud computing offers are not as relevant. In general, the industry sector and also critical infrastructure providers perceive cloud computing as fairly important (85 % on average - total value) for their businesses.

3) Information security requirements analysis: Critical infrastructures are at the fundament of today's societies as a consequence, failures and breakdowns may lead to serious repercussions. Hence it is important that software that is operated in the field of critical infrastructures is designed and built in a secure manner. The same concerns apply if the IT services are operated in cloud environments. In addition, we analyze the opinions regarding certain security attributes (availability, confidentiality, integrity, and auditing) for diverse IT services in a cloud environment as well as for generic cloud characteristics.

In general we found out that: (1) Information security is generally recognized as a very important matter by critical infrastructure providers. (2) Auditing is not perceived as important as availability, confidentiality, or integrity.

4) Security requirements for IT services: The following four common IT services were chosen for the security requirements elicitation:

- customer web platform
- enterprise management software (e.g. SAP)
- industrial control system / SCADA
- IT infrastructure (e.g. DNS, mail)

Our analysis shows that for the two sectors industry and critical infrastructure providers the smallest differences in information security requirements are for confidentiality and availability. The biggest difference was observed for integrity. Here the members of the industrial sector reported a higher-than-average need for security. In general the industry sector shows slightly higher information security needs than the critical infrastructure providers, as highlighted in Figure 3.

5) Security requirements for generic cloud computing characteristics: In this section the common NIST cloud characteristics on-demand self-service, broad network access, resource pooling, rapid elasticity, and measured service are evaluated for their required level of security for the two domains examined. Here the same results as determined in the previous investigation on the security requirements for IT services apply: In general the industry sector shows higher security needs than critical infrastructure providers (Figure 4). The highest requirements are specified for on-demand self-service.

6) Information security topics for a cloud migration guideline: The field information security consists of many controls that could be considered for hardening IT services. The relevance of several information security controls (risk assessment, incident response, SLA management, architectural patterns, service life cycle, socio-technical issues, autonomic security management, forensic and auditing, international standards) was asked for in the survey, shown in Figure 5. The outcome is again comparable with other questions from the survey, where

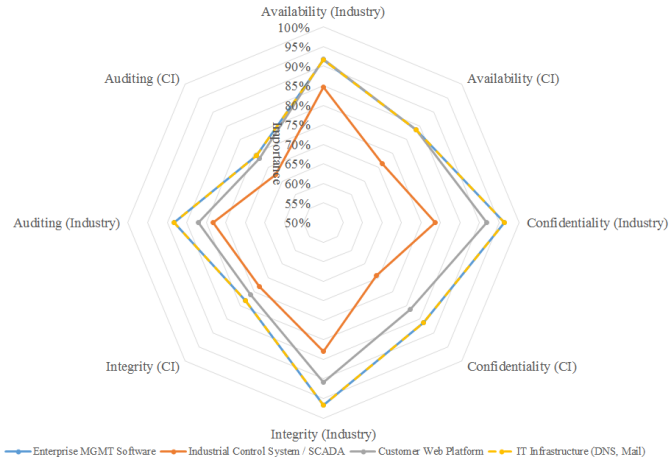

Fig. 3. Net chart of critical infrastructure providers' information security requirements for IT services compared to the industry domain. The industry sector in comparison generally has higher requirements.

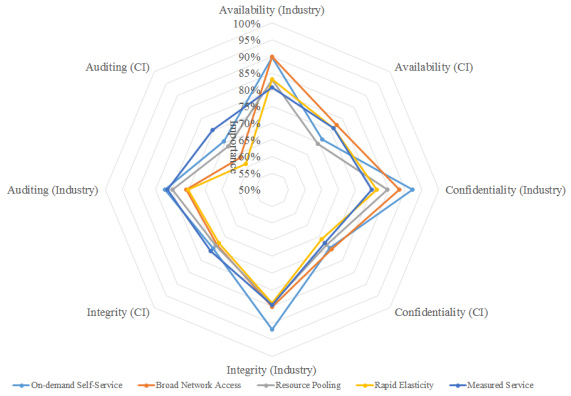

Fig. 4. Net chart of critical infrastructure providers' information security requirements for cloud computing characteristics compared to the industry domain. The industry sector in comparison generally has higher requirements.

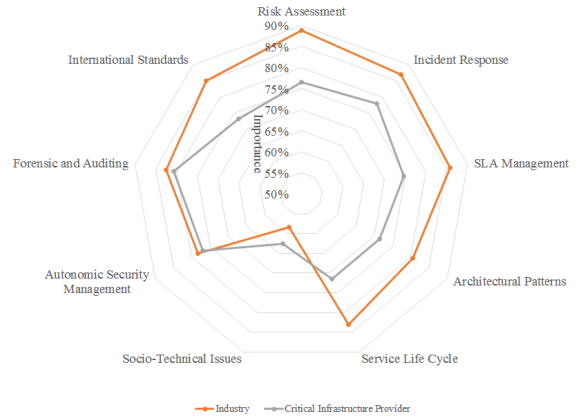

Fig. 5. Importance of information security topics for a cloud migration guideline.

the industry sector generally has a higher need for security as the critical infrastructure providers. Only socio-technical issues are slightly more important to critical infrastructure providers than to industry.

IV. PROCESS-BASED INFORMATION SECURITY MIGRATION GUIDELINE

In this section we, first of all, introduce an extensive set of security controls, which is based on the security topics that we addressed in our questionnaire (Chapter III). We use these security controls to build our taxonomy and use it for evaluating security related guidelines. Finally, based on the evaluation outcome we conclude and propose a process-based guideline(model) for secure service migration towards the Cloud environments.

A. Secure cloud migration taxonomy

The questionnaire results presented in Section III are used as a foundation for building the taxonomy that we present in this section. In our questionnaire we based the security related topics on the initial analysis of the security related aspects within the SECCRIT project[1] with respect to critical infrastructures. Within our taxonomy we now cover a more extensive set of security related topics. This extensive list of topics can be seen in Table III, where we have cross checked these topics with the state of the art security guidelines to investigate how and whether they are being addressed by each one of them.

The taxonomy depicted in Table III includes 34 security controls used for evaluating guidelines, by investigating how they address secure service migration in cloud based environments. Although, each individual security control covers a separate security dimension we use them in our taxonomy to answer the following:

1) *Is the security control enumerated and defined within the observed guideline?*
   Within this question the following security controls are covered: security requirements, privacy requirements, security architecture design, security risk assessment, threat management, vulnerability management, security testing, secure life cycle phases plan, development of security controls, data locality, incident handling, environment hardening, operational enablement, maturity levels, case studies, application migration.
2) *Is the security control implemented in a form of a process?*
   Within this question the following security controls are covered: security requirements, privacy requirements, security and privacy training, security risk assessment, threat management, security testing, regular improvement of security process artifacts, security life cycle phase plan, continuous monitoring of system and services, security planing for the project, integration of proposed concepts in established environments, privacy impact assessment, security accreditation/certification, information disposal, establishing trust strategies, data locality, legal compliance, incident handling, maturity levels, disaster recovery, consideration of security aspects in data migration, application migration.
3) *Does the security control involve architecture or conceptual design?*
   Within this question the following security controls are covered: security architecture design, secure planing for the project, disaster recovery, application migration.

[1]SECCRIT project, https://www.seccrit.eu

In Table III we detail our results of the analysis of guidelines that support or address migration of services towards Cloud-based environments.
Although being addressed by 60% of guidelines[26], [27], [28], [29], [30], [24] the first two controls, security and privacy requirements, are unfortunately either enumerating a narrow set of requirements or referencing a third party set. Most of the evaluated guidelines[26], [27], [30], [24] provide adequate approaches for increasing awareness for security and privacy in form of a training to support the enumerated requirements. However, only [26], [30], [24] provide concrete steps for accomplishing this requirement. Next security control is the security architecture design where we investigated the proposed architectural solutions and entailing processes, which were supported only by 50% of the guidelines[26], [27], [28], [30], [31] that we addressed. Whereby from these 50% only [26], [28] provide a generic solutions. Furthermore, security risk assessment was addressed by the majority of the evaluated guidelines where only 10% of the guidelines [32] have not included or considered it as relevant. However, only the following from the above mentioned guidelines detail the risk assessment approach [26], [27], [29], [24], others provide only a generic solutions. Although, threat management and vulnerability management are essential for implementing risk assessment, only 40% of guidelines[26], [29], [30], [24] support threat management whereas 50% of guidelines[26], [27], [30], [31], [24] support vulnerability assessment. In case of threat management only ENISA [29] is providing a generic solutions, whereas in case of vulnerabilities only NIST cyber security framework[24] is focusing only on generic solutions. Consideration of security practices during the development phase are unfortunately supported by only 20% of evaluated guidelines[26], [31] focusing on generic solutions, where as performing security tests was covered by only 50% of guidelines[26], [27], [29], [30], [31] most of them providing a detailed approach for tests handling. Improvement of security related processes is a continuous requirement which was supported by 50% of guidelines[26], [27], [31], [24], [33]. Formulating structured life-cycle phases for performing certain actions or tasks was embraced by 50% of the guidelines[26], [27], [30], [31], [34]. Processes for delivering continuity in terms of monitoring systems or services was defined by 60% of evaluated guidelines[26], [27], [28], [29], [24], [33]. Furthermore, only 50% of evaluated guidelines[26], [27], [28], [30], [33] implement security planning as process. The evaluation

shows that only CSA[31] advises in their guideline how to integrate the proposed concepts.

The assessment of privacy concerns was covered by the minority, only 30% of guidelines[26], [27], [28]. We investigated whether the guidelines support development of security related controls but unfortunately it was addressed only by only 40% of evaluated guidelines. The NIST SP800-64[27] was the only guideline interested in accreditation or certification processes and information service disposal. Suggesting processes or models related with establishing trust was a topic addressed only by 20% of guidelines[27], [34] where Microsoft[34] proposed a solution on a use case scenario for their Windows Azure. A very important security control which was also considered in our Section III-B is geographical location of the data addressed by only 30% of guidelines[28], [33], [34].

Legislative requirements were one of a most referred points which were addressed through compliance and legislative requirements from 70% of guidelines[27], [28], [29], [30], [31], [33], [34]. However, due to the area-specific and often nationally bound nature of legal requirements, it cannot be expected from these guidelines to provide universal and sufficient guidance here. Solutions for handling incidents was proposed by 60% of guidelines[26], [27], [28], [29], [31], [24]. The Software Assurance Maturity Model (SAMM)[30] was the only guideline concerned with hardening and operational enablement. Only 20% of guidelines[32], [30] proposed and define maturity levels in their work. The Software Assurance Maturity Model (SAMM)[30] and Microsoft [34] supported their guidelines with a use case. Disaster recovery solutions and concepts were proposed from 40% of the guidelines[27], [32], [34], [30]. Consideration of security aspects during migration is only addressed by 30% of the guidelines[33], [31], [32]. Finally, only 30% of the guidelines[33], [34], [32] address and propose application migration concepts or processes which are tightly related with the technology the guideline was made for.

The results from our detailed evaluation show that Microsoft SDL[26], NIST SP 800-64 [27], and Software Assurance Maturity Model [30] are the guidelines that fulfill most of the security related controls, Table III Coverage of security controls per guideline, that we used in our taxonomy. Not all of the controls have been covered by at least one of the guidelines. Therefore, we propose a comprehensive solution for handling such a scenario in the following section.

B. Secure cloud migration life cycle

To use such a taxonomy in an effective way, it should be incorporated into a process that gives attention to the information security aspects of *cloudification*.

According to our literature review there is currently no security development life cycle that explicitly takes into account a cloud migration scenario. We therefore propose a novel approach for a Cloudification Security Development Life cycle (CloudSDLv1) of IT services which we base on common security development life cycles [26], [27], [30]. Our approach for CloudSDLv1 is shown in Figure 6. It is built around the security requirements relevant to the *cloudified* product.

We consider the following use cases for CloudSDLv1:

- Software development for cloud environment from scratch.
- Software migration from legacy system to cloud (adoption for cloud).
- Software migration from private to public cloud and vice versa.

CloudSDLv1 comprises five phases:

1) Analysis: In this phase a decision is made upon which service or which part of a service is to be migrated to cloud. The IT service that is to be *cloudified* is analyzed for cloud fitness and the initial set of security requirements is specified. Ideally, if security requirements for the IT service already exist they have to be taken into account and if needed adopted to the new circumstances. In particular, this should also include security requirements indirectly resulting from the *cloudification* of a certain service. For example, this might refer to novel needs for providing credible digital evidence on the providers' security-related conduct for the potential case of legal conflicts or to cloud-specific requirements from data protection law. In this phase we also suggest to analyze which implications the *cloudification* of the IT service has on the organization and the business. Any implications on information security have also to be converted into security requirements.

2) Design: In the design phase the software architecture for the to-be-migrated IT service is constructed on the basis of the security requirements specified in the analysis phase. If necessary refinements to the security requirements are made.

3) Implementation: Based on the design the software is implemented. If necessary refinements to the security requirements are made. Additionally in this phase the organization is, where necessary, prepared for the use of the *cloudified* IT service.

4) Verification: In the verification phase the software is tested against the specified security requirements. Also the readiness of the organization for the *cloudified* IT service is verified.

5) Deployment: In this final phase of the CloudSDLv1 the IT service is deployed on the cloud environment, taking into account the security requirements related to platform configuration.

V. Conclusion

In this paper we have show that the major difference in importance of information security topics between industrial stakeholders and critical infrastructure provider is geolocation. This means that storing data within the same legal domain is more important for critical infrastructure provider than for industrial stakeholder, other than that interests are aligned. We have used this information to survey existing industrial and critical infrastructure guidelines to update our initial set of security controls and fed this into a proposal for a cloudification guideline for critical infrastructure providers. We present a novel approach for a process-based information security guideline. The presented taxonomy together with

TABLE III
STANDARDS AND GUIDELINES FOR MIGRATING CRITICAL INFRASTRUCTURE SERVICES TO TAXONOMY

SECURITY CONTROLS	Microsoft: Security development lifecycle	NIST SP800-64: Security Considerations in the System Development Life Cycle	NIST SP 800-144: Guidelines on Security and Privacy in Public Cloud Computing	ENISA: A CIIP perspective on cloud computing services	OPENSAMM: A guide to building security into software development	CSA: Security guidance for critical areas of focus in cloud computing	NIST: Framework for Improving Critical Infrastructure Cybersecurity	Migrating your Existing Applications to the AWS Cloud	Moving Applications to the Cloud on Microsoft Azure	Cloud migration research: A systematic review	Control implemented in Percentage of Guidelines
Security Requirements	x	x	x	x	x		x				60%
Privacy Requirements	x	x	x	x	x		x				60%
Security& Privacy Training	x	x			x		x				40%
Security Architectural Design	x	x	x		x	x					50%
Security Risk Assessment	x	x	x	x	x	x	x	x	x		90%
Threat Management	x			x	x		x				40%
Vulnerability Management	x	x			x	x	x				50%
Secure Coding Practices	x					x					20%
Security Testing	x	x		x	x	x					50%
Regular Improvement of Security Process Artefacts	x	x				x	x	x			50%
Security Life Cycle Phases Plan	x	x			x	x			x		50%
Continuous monitoring of systems and services	x	x	x	x			x	x			60%
Security Planning for the Project	x	x	x		x			x			50%
Integration of proposed Concepts in Established Environments						x					10%
Privacy Impact Assessment	x	x	x								30%
Development of Security Controls		x				x	x	x			40%
Security Accreditation/Certification		x									10%
Information Disposal		x									10%
Establish trust strategies			x						x		20%
Data locality			x					x	x		30%
Legal compliance		x	x	x	x	x		x	x		70%
Incident Handling	x	x	x	x		x	x				60%
Environment Hardening					x						10%
Operational Enablememt					x						10%
Maturity Levels					x					x	20%
Case Studies					x				x		20%
Disaster Recovery		x				x	x			x	40%
Consideration of Security Aspects in Data Migration						x		x		x	30%
Application Migration								x	x	x	30%
Coverage of Security Controls:	52%	62%	38%	28%	52%	48%	38%	31%	24%	14%	

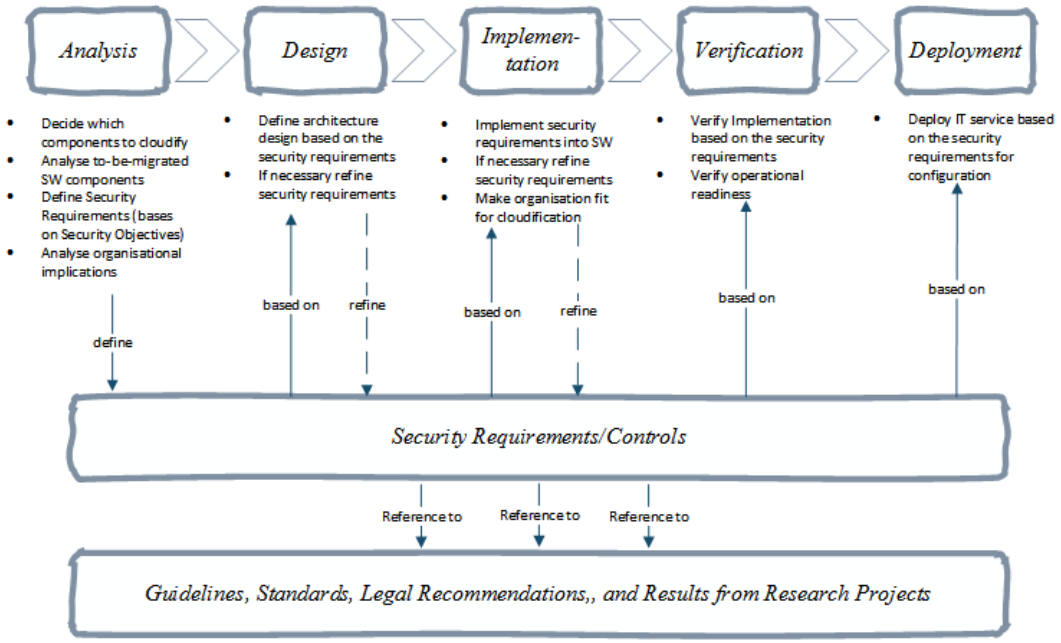

Fig. 6. Process-based information security guideline for cloud migration

the proposed *cloudification* security development life cycle (CloudSDLv1) will support critical infrastructure providers in migrating their legacy IT services to the cloud. Based on this work our next steps will be a) an extension of the presented taxonomy towards research results of the EU FP7 research project SECCRIT, and b) empirical evaluation of our taxonomy and CloudSDLv1 in a real world scenario.

ACKNOWLEDGEMENTS

The research presented in this paper has been funded by the European Union (H2020 project PRISMACLOUD, Grant No. 644962 and FP7 project SECCRIT, Grant No. 312758).

REFERENCES

[1] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," *Future Generation Computer Systems*, vol. 25, no. 6, pp. 599–616, 2009. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X08001957

[2] R. Buyya, "Market-Oriented Cloud Computing: Vision, Hype, and Reality of Delivering Computing As the 5th Utility," in *Proceedings of the 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid*, ser. CCGRID '09. Washington, DC, USA: IEEE Computer Society, 2009. [Online]. Available: http://dx.doi.org/10.1109/CCGRID.2009.97

[3] P. Mell and T. Grance, "The NIST definition of cloud computing," *National Institute of Standards and Technology*, vol. 53, no. 6, p. 50, 2009.

[4] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica *et al.*, "A view of cloud computing," *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, 2010.

[5] K. Sun and Y. Li, "Effort Estimation in Cloud Migration Process," in *Service Oriented System Engineering (SOSE), 2013 IEEE 7th International Symposium on*, March 2013, pp. 84–91.

[6] A. Khajeh-Hosseini, I. Sommerville, J. Bogaerts, and P. Teregowda, "Decision Support Tools for Cloud Migration in the Enterprise," in *Cloud Computing (CLOUD), 2011 IEEE International Conference on*, July 2011, Service migration, pp. 541–548.

[7] P. Jamshidi, A. Ahmad, and C. Pahl, "Cloud Migration Research: A Systematic Review," *Cloud Computing, IEEE Transactions on*, vol. 1, no. 2, pp. 142–157, July 2013.

[8] Q. Zhang, L. Cheng, and R. Boutaba, "Cloud computing: state-of-the-art and research challenges," *Journal of internet services and applications*, vol. 1, no. 1, pp. 7–18, 2010.

[9] T. Dillon, C. Wu, and E. Chang, "Cloud Computing: Issues and Challenges," in *Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on*, April 2010, pp. 27–33.

[10] S. Kaisler and W. Money, "Service Migration in a Cloud Architecture," in *System Sciences (HICSS), 2011 44th Hawaii International Conference on*, Jan 2011, Service migration, pp. 1–10.

[11] A. Khajeh-Hosseini, D. Greenwood, and I. Sommerville, "Cloud Migration: A Case Study of Migrating an Enterprise IT System to IaaS," in *Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on*, July 2010, Service migration, pp. 450–457.

[12] C. Fehling, F. Leymann, S. Ruehl, M. Rudek, and S. Verclas, "Service Migration Patterns – Decision Support and Best Practices for the Migration of Existing Service-Based Applications to Cloud Environments," in *Service-Oriented Computing and Applications (SOCA), 2013 IEEE 6th International Conference on*, Dec 2013, Service migration, pp. 9–16.

[13] C. Alcaraz and S. Zeadally, "Critical infrastructure protection: Requirements and challenges for the 21st century," *International Journal of Critical Infrastructure Protection*, no. 0, pp. –, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1874548214000791

[14] S. Subashini and V. Kavitha, "A survey on security issues in service delivery models of cloud computing," *Journal of Network and Computer Applications*, vol. 34, no. 1, pp. 1–11, 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804510001281

[15] B. Kandukuri, V. Paturi, and A. Rakshit, "Cloud Security Issues," in *Services Computing, 2009. SCC '09. IEEE International Conference on*, Sept 2009, Security, pp. 517–520.

[16] D. Zissis and D. Lekkas, "Addressing cloud computing security issues," *Future Generation Computer Systems*, vol. 28, no. 3, pp. 583–592, 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X10002554

[17] R. Piggin, "Are industrial control systems ready for the cloud?" *International Journal of Critical Infrastructure Protection*, no. 0, pp. –, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1874548214000821

[18] D. Chen and H. Zhao, "Data Security and Privacy Protection Issues in Cloud Computing," in *Computer Science and Electronics Engineering (ICCSEE), 2012 International Conference on*, vol. 1, March 2012, Security, pp. 647–651.

[19] M. Jensen, J. Schwenk, N. Gruschka, and L. Iacono, "On Technical Security Issues in Cloud Computing," in *Cloud Computing, 2009. CLOUD '09. IEEE International Conference on*, Sept 2009, Security, pp. 109–116.

[20] M. Nanavati, P. Colp, B. Aiello, and A. Warfield, "Cloud Security: A Gathering Storm," *Commun. ACM*, vol. 57, no. 5, pp. 70–79, May 2014. [Online]. Available: http://doi.acm.org/10.1145/2593686

[21] M. Y. A. Younis and K. Kifayat, "Secure cloud computing for critical infrastructure: A survey," *Liverpool John Moores University, United Kingdom, Tech. Rep*, 2013.

[22] S. Paudel, M. Tauber, C. Wagner, A. Hudic, and W.-K. Ng, "Categorization of standards, guidelines and tools for secure system design for critical infrastructure it in the cloud," in *Cloud Computing Technology and Science (CloudCom), 2014 IEEE 6th International Conference on*, Dec 2014, pp. 956–963.

[23] M. Dekker, "Critical Cloud Computing:CIIP Perspective on Cloud Computing," 2013, [Online; accessed 19-July-2013].

[24] N. I. of Standards, T. (NIST), and U. S. of America, "Framework for improving critical infrastructure cybersecurity," 2014.

[25] J. Morin, J. Aubert, and B. Gateau, "Towards cloud computing SLA risk management: issues and challenges," in *System Science (HICSS), 2012 45th Hawaii International Conference on*. IEEE, 2012, pp. 5509–5514.

[26] M. Howard and S. Lipner, *The security development lifecycle*. O'Reilly Media, Incorporated, 2009.

[27] R. Kissel, K. M. Stine, M. A. Scholl, H. Rossman, J. Fahlsing, and J. Gulick, "SP 800-64 Security Considerations in the System Development Life Cycle," 2008.

[28] W. Jansen, T. Grance *et al.*, "Guidelines on security and privacy in public cloud computing," *NIST special publication*, vol. 800, p. 144, 2011.

[29] M. Dekker, "Critical Cloud Computing-A CIIP perspective on cloud computing services," *white paper, December*, 2012.

[30] P. Chandra, "The Software Assurance Maturity Model - A guide to building security into software development," 2009.

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

[32] P. Jamshidi, A. Ahmad, and C. Pahl, "Cloud migration research: a systematic review," *Cloud Computing, IEEE Transactions on*, vol. 1, no. 2, pp. 142–157, 2013.

[33] J. Varia, "Migrating your existing applications to the aws cloud," *A Phase-driven Approach to Cloud Migration*, 2010.

[34] D. Betts, A. Homer, A. Jezierski, M. Narumoto, and H. Zhang, "Moving applications to the cloud on microsoft azure," *MSDN Library*, 2012.

A Multi-Layer and Multi-Tenant Cloud Assurance Evaluation Methodology

The paper "A Multi-Layer and Multi-Tenant Cloud Assurance Evaluation Methodology" was published on IEEE International Conference on Cloud Computing Technology and Science (CloudCom) 2014 in Singapore.

The paper can be found online under the following link: `http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=7037693`

A Multi-Layer and Multi-Tenant Cloud Assurance Evaluation Methodology

Aleksandar Hudic, Markus Tauber
Austrian Institute of Technology
{aleksandar.hudic, markus.tauber}
@ait.ac.at

Maria Krotsiani, George Spanoudakis
City University London
{g.e.spanoudakis, maria.krotsiani.1}
@city.ac.uk

Mauthe, Andreas
Lancaster University
<a.mauthe@lancaster.ac.uk>

***Abstract*— Data with high security requirements is being processed and stored with increasing frequency in the Cloud. To guarantee that this data is being dealt with a secure manner we investigate the applicability of Assurance methodologies. In a typical Cloud environment the setup of multiple layers and different stakeholders determines security properties of individual components that are used to compose Cloud applications. We present a methodology adapted from Common Criteria for aggregating information regarding the security properties of individual constituent components of Cloud applications. This aggregated information is used to categorise the overall application security, in terms of Assurance Levels and to provide a continuous assurance level evaluation.**

Keywords— critical infrastructures, assurance, cloud computing, public safety,

I. INTRODUCTION

Rapid propagation of Cloud Computing paradigm across distinct domains and organisations reflected on IT systems and services through a large transformation. Hence the service model is commoditised and delivered in a manner similar to utilities such as water, electricity, gas, and telephony. The main interest for such adoption by organizations is to increase efficiency and minimize IT costs by offering new concepts such as elasticity, scalability and on-demand resource provisioning. However, in order to automatically provision resources for elastically adaptive Cloud applications it requires both the applications and the underlying platform to be constantly monitored to capture information at various levels and time intervals. This is particularly manifested in Critical Infrastructures, which require even more attention when these systems are hosted on top of Cloud environments.

However, the use of cloud computing has introduced many new risks that must be comprehended before an organisation is considering adopting the cloud and when using cloud services. Moreover, due to the complexity of the application execution environment, routine tasks such as monitoring or performance analysis become quite complex. These tasks often require close interaction and assessment in different layers of the cloud stack. For example, when a distributed application that has been provisioned on a cluster a virtual machine (VM) needs to provide assurance or even has to be certified for a specific security property This might require to also monitor the execution of the application on the VMs, as well as the availability of physical resources of the VMs. Thus, this would require the use of different sets of tools to collect and analyse the performance of data from each level in order to reach the point to certify the application.

Under these circumstances, we should gather different types of information of different granularity, from low-level system metrics (e.g. CPU usage, network traffic, memory allocation, etc.) to high-level application specific metrics (e.g. throughput, latency, availability, etc.). These are collected across multiple levels (physical, virtualization, application level) in a Cloud environment at different time intervals. Hence, the challenge in this case is to define a way to aggregate these different types of information from different levels in order to provide an overall assurance, and determine how changes in individual assurance level of every component affects the overall assurance.

In this paper, we propose a comprehensive concept for assessing security properties across multiple layers, with different stakeholders, for composite based systems. The assessment concept is aligned with the customisable set of security policies, which can be flexibly adopted according to various use case requirements, to derive evaluation of every individual component of a service or a system.

The rest of the paper is structured as follows. Section II outlines the work related towards our concept. Section III describes our approach and introduces the Assurance Assessment Model, the way we define assurance levels, how we abstract the service as a general tree and the assurance aggregation process. In Section IV the evaluation of the approach is provided based on a Use Case Scenario. The last but not the least, section V provides concluding remarks and directions for future work.

II. RELATED WORK

Traditional approaches for assurance assessment in the cloud, such as Cloud Security Alliance (CSA) [12], Information Technology Assurance Framework (ITAF) [17], or Cloud Computing Information Assurance Framework from ENISA [18], usually build on existing frameworks such as ISO/IEC 27000-series (e.g. current work in progress ISO/IEC 27017 and ISO/IEC 27018, which are focusing on information security and data protection in Cloud), PCI DSS Cloud Guideline [13], COBIT [14], NIST [16], or IT Baseline Protection Catalogues [15].

Our latest state of the art research [10] looks at an existing approaches, namely the Common Criteria framework [6] for assurance of IT systems as it is the most dominant work in the field. However, this comprehensive framework is focused

mostly on assessing assurance in the development phase of the life cycle, and lacks the support in the following production phase.

Unlike traditional approaches, the work derived from Krotsiani et.al. [11] proposes a novel approach for certifying the security of cloud services based on incremental certification of security properties for different types of cloud services, including IaaS, PaaS and SaaS services. This approach uses operational evidence from provisioning of such services through continuous monitoring. Although the model doesn't directly address assurance as a prior objective, it can be adopted to efficiently assess assurance at various levels and time intervals.

Our approach is related to autonomic monitoring systems, which is based on the SECCRIT architecture model [7] and on an evidence-gathering model [8] for assurance assessment in critical infrastructures hosted on top of cloud environments. Hence, we emphasize the importance of observing systems in their production phase, as well as their dependencies with other corresponding elements inside of heterogeneous systems. It should be noted that our work is a part of a broader research programme, undertaken by the EU F7 project SECCRIT [4].

III. MULTI LAYER ASSURANCE ASSESMENT MODEL

The most commonly referred cloud architecture model for distributed service provisioning is defined by National Institute of Standards and Technology (NIST) [3]. This NIST model depicts the cloud architecture through a dynamic tree-layered service-provisioning model (infrastructure, platform and software - as a Service layer), capable of scaling services across distinct administrative and legislative domains. However, the common practice for provisioning and delivering services, as well as the abstraction of those layers and driven technologies differentiate based on the business objectives of a particular cloud provider. Hence, the traditional assessment frameworks (e.g. COBIT, ISO 27000 series) are not really applicable, especially when addressing security related concerns in cloud environments, as seen in [10].

However, in order to build a comprehensive and flexible framework that is able to acquire heterogeneous information across the cloud stack, the following objectives have to be addressed:

- cross layer assessment
- technology independence
- information acquisition restrictions
- assessment, quantification and aggregation of different information sets

The assessment of such services (when taking in to account different layers of clouds with different stakeholders, different business and security objectives, high degree of service complexity, business model, and distinct technologies) requires a compact solution, able to embrace all requirements and produce an effective assessment tool. Hence, we adopt Common Criteria [6] for addressing assurance in cloud related environments. Although, Common Criteria offers a comprehensive solution for assurance assessment, it lacks the support in the production phase, especially when referring to those services that are hosted on top of the cloud architectures. Taking this into account and the above mentioned objectives, we use the fundaments of the Common Criteria approach in order to address assurance assessment of complex services hosted in cloud infrastructures. Furthermore, the policies of some cloud providers restrict information crawling across their cloud stack (for instance software as a service cloud provider will hesitate to reveal the information of underlying service being provided, in order to mitigate potential attack vectors on its infrastructure). Therefore, it is harder to analyse and indicate or predict any security situation in such environments. Hence, we distinguish two main categories: a) solutions based on open-source cloud environments (i.e. solutions where we are able to freely acquire necessary information without restrictions); and b) closed cloud environments with restricted information access (i.e. public cloud providers which provide any additional information via the Service Level Agreements (SLA) [21][22]). Due to the flexibility of acquiring the information and ability to modify services for provisioning the information, in this paper we focus primarily on open-source cloud solutions (e.g. OpenStack [23], CloudStack [24]). This does, however, not limit our approach to these environments.

The assessment and aggregation of different information sets (i.e. analysis of a particular entity in the cloud in respect to a certain set of properties) is derived via the assurance levels, supported with aggregation policies (i.e. decision making algorithms that cluster the security properties of each class towards the predefined assurance levels), aligned with the Common Criteria approach [6].

A. *Assurance assesment model*

Considering the objectives discussed above and building on the research presented in [10] we propose a comprehensive and flexible approach for performing Assurance assessment. The approach is using a well-defined set of security properties, provided by the CUMULUS project [5], which are additionally aligned with the SECCRIT vulnerability catalogue [20] and the notorious nine from Cloud Security Alliance [19].

Our assessment model emphasises three core assessment entities: *Target of Evaluation* (ToE), *Group of Evaluation* (GoE) and *Component of Evaluation* (CoE). These entities are aligned with the Common Criteria assessment framework, and are therefore designed to offer flexibility, determination of the precise impact of the individual components or group of components, scalability of assessment across different time intervals, and the possibility to highlight each individual entity of the system as an independent point of evaluation. Furthermore, we designed our model into a hierarchical tree structure, defined with parent-child object relationship. Each parent can be in direct relationship with multiple child objects. The parent object that does not have any related child objects is referred to as a *leaf* object. Additionally, we also define *associations*, *dependencies*, *associated component sets* and *assurance profiles*, as supporting assessment elements of the ToE in our model. Figure 1 illustrates the fundamental

elements of the Assurance Assessment Model of our approach. More specifically, it presents a way a particular service can be abstracted through a set of hierarchically organized components. We use these abstraction elements to build our model and to efficiently assess assurance according to a predefined set of security properties derived from CUMULUS project.

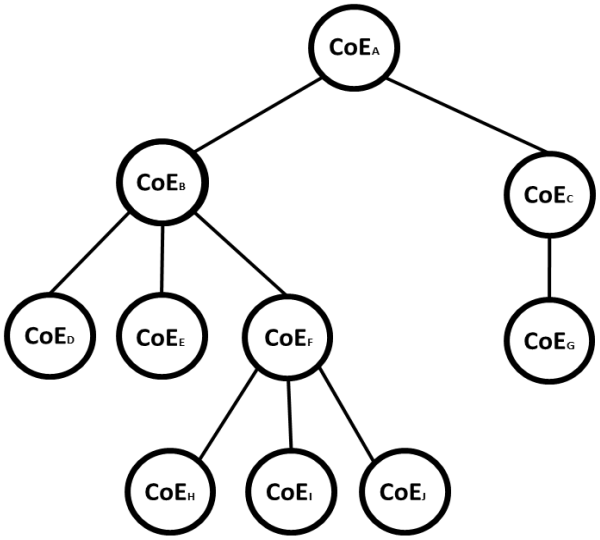

Figure 1: Hierarchical illustration of services via the general tree model structure. We define the service or application as a Target of Evaluation (ToE) depicted with the individual Components of Evaluation (CoE), whereby each individual CoE can be associated with N distinct CoEs, referred as Associated Component Set (ACS). The correlation between two individual CoEs is referred as a Component Dependency that is a formal compound of Association. Moreover, we group CoEs in order to arbitrarily elaborate assurance of components in respect to certain security classes, these groups are then formally defined as Groups of Evaluation (GoE).

The initial step of our assessment model defines and details the ToE. This can be either an asset of the cloud referred to as service (e.g. a specific service operation, a set of service operations, data managed by the service) or an asset that is required or contributes to the realization of a cloud service (e.g., a virtual machine). Moreover, each ToE contains a set of attributes such as: (i) security objectives, which are mapped towards the related set of security claims and are formally referred to as *Security Properties* (SP); (ii) attributes that define the type of assurance (e.g. information or system assurance) according to the assurance model presented in [10]; (iii) a short description of the ToE; and (iv) the assessment interval. The security objectives are the statements of intent to counter the identified threats by IT measures. Each ToE can be formally defined as ToE $\equiv \top =$ {COE$_i$, $i \in$ **N**} | {GOE$_i$, $i \in$ **N**}. This generalized statement in respect to Figure 1 would be formulated as ToE $\equiv$ COE$_A$ = {COE$_i$, $i \in$ ⟨B, C, D, E, F, G, H, I, J⟩}. Group of objects, formally referred as Group of Evaluation (GoE) and defined as GoE = {CoEi, i ∈ N}, are a compound set of individual objects that share common properties based on which the assessment is conducted. If we refer in particular to Figure 1, GoE can be formulated as compound of objects, e.g. GOE$_1$ = {COE$_i$, $i \in$ ⟨F, H, I, J⟩}. Each individual object to which we refer to as the component of evaluation (CoE) can be also handled as an independent ToE. Each GoE is composed of (i) attributes, used for describing a particular group; (ii) assurance profile, which is the essential element for evaluation; (iii) associations, an element used to describe relationships between different groups in the scope of the evaluated target; and (iv) individual components.

Component Dependency (CD) is a correlation between two individual components of the evaluated system (i.e where CD$_{ij}$ {⟨COE$_i$, COE$_j$⟩, $i, j \in$N}), that arises when a component is not self-sufficient and relies upon the presence of another component, e.g. when referring to Figure 1 CD$_{CG}$={COE$_C$, COE$_G$}. *Association* is a set of two individual components that are in a direct parent-child relationship with a defined dependency, for which it is valid: $\forall$ AS$_i$ $i \in$ **N** $\equiv$!$\exists$ CD$_{ij}$ { ⟨COE$_i$, COE$_j$⟩, $i, j \in$N} $\Longrightarrow$ COE$_i$ parent of COE$_j$. An individual parent object can be associated with N distinct child objects, which we formally refer to as *Associated Component Set* (ACS), for which the following statements are valid: ACS$_K$ ≔ ACS (COE$_K$) = {COE$_i$, $i \in$ **N**}, $\forall$ COE$_i$ $\Longrightarrow$!$\exists$ Parent = COE$_K$ and $\nexists$ CD$_{ij}$ {⟨COE$_i$, COE$_j$⟩, $i, j \in$ N}.

Finally, the last element of our assessment model is the *Assurance profile* (AP), which is an essential element used to define policy related with security properties that are mapped to the Assurance classes (AC) of a particular CoE or GoE. These security properties will at the end define the level of assurance for an individual component, group or even a whole system. We emphasize two types of Assurance profiles setup: Uniform Assurance Profile (AP$_U$), which is always the same regardless of class, evaluated object, group or target; and Custom Assurance Profile (AP$_P$), which can be customised depending on the object of appliance. In Table 1 we illustrated the AP$_U$ for a particular assurance class. Furthermore, we can also assign a custom Assurance Profile to a particular CoE, GoE or ToE.

B. Assurance Levels

Assurance level (AL) outlines the scale of measurement for evaluating predefined ToE, GoE or CoE, where every individual CoE or GoE contributes directly to the assurance level of the ToE, by meeting a set of SPs (i.e. certain set of security criteria). Moreover, the SPs derive the AL per individual AC, by taking also into consideration the dependencies of the evaluated object (e.g. component, group or target of evaluation) if such are present. However, each AC may contain *k* of SP (*k* number of SPs) as shown in equations 5 and 6. Due to the binary decision making concept that we apply in our approach, there can be 2^k combinations of distinct SP states, where $2^k > N$, and N is the cardinality of AL, in terms of security properties (AL= {1, 2, 3, 4 … N}). Therefore each individual combinations of SPs {SP$_1$, SP $_2$, SP$_3$, SP$_4$ … SP$_N$}, associated with a particular AC, are formally referred as Security Property Vector (SPV) (equations 3, 4, 5, 6). Security Property Vector defines the current state of an object by

identifying particular set of security properties. Each SPV, is associated with a particular assurance class, whereby each class can comprise multiple SPVs. Thus, in order to scale 2^k states over N assurance levels, we encode ranges in hexadecimal vectors that clusters potential set of states that correspond to a particular level, as shown in Table 1. Hence, each individual AL is assigned with multiple SPVs, which are formally referred as Vector Set (VS), (equation 2).

Table 1 presents an example of Assurance profile for a particular Assurance class. More specifically, it illustrates a set of relevant SPs clustered per individual AC_K represented with a hexadecimal vector. The left hand side of the table shows the SPVs, sorted by relevance, and all potential combinations for a particular security vector SPV = [SP_4, SP_3, SP_2, SP_1]. The right hand side shows a binary vector for AL_i ($i \in \{1, 2, 3 \dots 7\}$), which associates particular set of SV vectors. At the bottom of the table the Hexadecimal representation of each particular binary AL vector is illustrated.

Table 1: Assurance level association for a particular Assurance class. Set of relevant SPVs clustered per individual AC_K represented with a hexadecimal vector. The left hand side of the table shows the SPVs, sorted per relevance, and all potential appearance combinations for a particular vector SPV = [SP_4, SP_3, SP_2, SP_1]. The right hand side shows a binary vector for AL_i, i ∈ {1, … 7} which associates particular set of SV vectors. At the bottom of the table the Hexadecimal representation of each particular binary AL vector is illustrated.

Security Property Vector (SPV)				**Assurance level association**							
SP_4	**SP_3**	**SP_2**	**SP_1**	**AL**	**AL_1**	**AL_2**	**AL_3**	**AL_4**	**AL_5**	**AL_6**	**AL_7**
0	0	0	0	-	0	0	0	0	0	0	0
0	0	0	1	**AL_1**	**1**	0	0	0	0	0	0
0	0	1	0	**AL_2**	0	**1**	0	0	0	0	0
0	0	1	1	**AL_2**	0	**1**	0	0	0	0	0
0	1	0	0	**AL_3**	0	0	**1**	0	0	0	0
0	1	0	1	**AL_3**	0	0	**1**	0	0	0	0
0	1	1	0	**AL_4**	0	0	0	**1**	0	0	0
0	1	1	1	**AL_4**	0	0	0	**1**	0	0	0
1	0	0	0	**AL_5**	0	0	0	0	**1**	0	0
1	0	0	1	**AL_5**	0	0	0	0	**1**	0	0
1	0	1	0	**AL_6**	0	0	0	0	0	**1**	0
1	0	1	1	**AL_6**	0	0	0	0	0	**1**	0
1	1	0	0	**AL_7**	0	0	0	0	0	0	**1**
1	1	0	1	**AL_7**	0	0	0	0	0	0	**1**
1	1	1	0	**AL_7**	0	0	0	0	0	0	**1**
1	1	1	1	**AL_7**	0	0	0	0	0	0	1
Hexadecimal AL vector					0002	000C	0030	00C0	0300	0C00	7000

$$\forall AL_K \in AC_X: \ !\exists \ VS, \quad \textbf{(1)}$$

$$VS = \{SPV_1, SPV_2 \dots SPV_N\}, \quad \textbf{(2)}$$

$$SPV_i = [\ SP_1, SP_2, SP_3, SP_4],\ SP_i = \{0,1\} \quad \textbf{(3)}$$

$$\forall VS \in \ AL_K : \ \exists \ SPV_i,\ i \in \mathbb{N} \quad \textbf{(4)}$$

$$\forall \ SPV_i \in AC_X: |SPV_i| = k \quad \textbf{(5)}$$

$$AC_X = \{SPV_1, SPV_2, SPV_3, \dots SPV_n\} \quad \textbf{(6)}$$

$$\bigcap_{i=1}^{i=n} AC_i = \emptyset \quad \textbf{(7)}$$

For each individual AC that is associated with a set of SPVs particular SP, inside of SPV, may vary. Nevertheless, every individual AC, regardless of the SPs, always has to have the same cardinality k (equation 5). In order to efficiently aggregate the assurance across the variety of architectural layers, ACs first has to fulfil the equations 5 and 6, stating that regardless of the AC, none of the SPs can be associated with more than one AC (equation 7).

Although, we abstract ALs over N levels, for the purpose of our empirical evaluation we will conduct the assessment over 7 ALs, therefore having minimum 3 SP per AC to be able to map all assurance levels with SPVs. Depending on the property set that a particular entity (i.e. class component, group or even a whole target of evaluation) is assigned with and due to the dynamic behaviour of the cloud the AL will also be dynamic and vary. Hence, it is highly important to efficiently assess the assurance in continuous manner, without being invasive on the performance of the service that is being evaluated or collocated.

C. Assurance Aggregation

As mentioned above, we propose a concept for the assurance aggregation through a recursive process, which aggregates the individual assurance levels of the underlying associated objects (i.e. it calculates the overall assurance of the components that are associated with the root component). If we illustrate a service through the illustrational model depicted in Figure 1, by conducting the proposed algorithm described in Figure 4 we can derive the overall assurance. Therefore by referring to Figure 1, we state the CoE_A as the ToE. Since, the CoE_A is associated with two additional components, CoE_B and CoE_C, which represent the associated components set (ACS_A) of the CoE_A and are additionally connected with other components. The overall assurance in this case has to be recursively aggregated from the leafs of the tree (i.e. by

$$ACS_{AL} = \textcircled{|}AC_X(SPV_i),\ AC_X \in CoE_M, i \in \{1\dots N\} \quad \textbf{(8)}$$

$$ACS_{AL}(i) \vdash DAL_{VS}(i) \quad \textbf{(9)}$$

$$AL_{VS} \subseteq DAL_{VS} \quad \textbf{(10)}$$

$$(DAL_{VS}(i) \wedge AL_{VS}(i)) \Rightarrow AL(AC_X)=i,\ AC_X \in CoE_M \quad \textbf{(11)}$$

$$!\exists\ AL_i \vDash \forall Min(CAL_j)\ \ i \in \{1\dots7\}, j \in \{1\dots N\} \quad \textbf{(12)}$$

aggregating all ACS (ACS_B, ACS_C and ACS_F). Therefore we will use tree traversal post order method to iteratively walk through the tree. For the scope of the first use case, we just refer to the concept of the tree traversal post order method as a tool for our concept. We slightly extend this method by integrating our Assurance Level Calculation Procedure (ALCP) from Figure 3 to recursively aggregate assurance.

The assurance level of the referenced ACS (ACS_F, ACS_B and ACS_C, respectively), by applying the ALCP aligned with the equation 8. The procedure sequentially conducts bitwise conjunction of individual SPs for each CoE for each ACS. Depending on the result of conjunction (1 or 0), we decide if we are discarding all SPVs with the bit that matches the result of the conjunction. For example, by discarding certain SPVs we are indirectly discarding those ALs that are not fulfilling the current set of SPs for particular ACS. The next step is to map the suitable ACS_{AL}, according to the Table 2, towards the

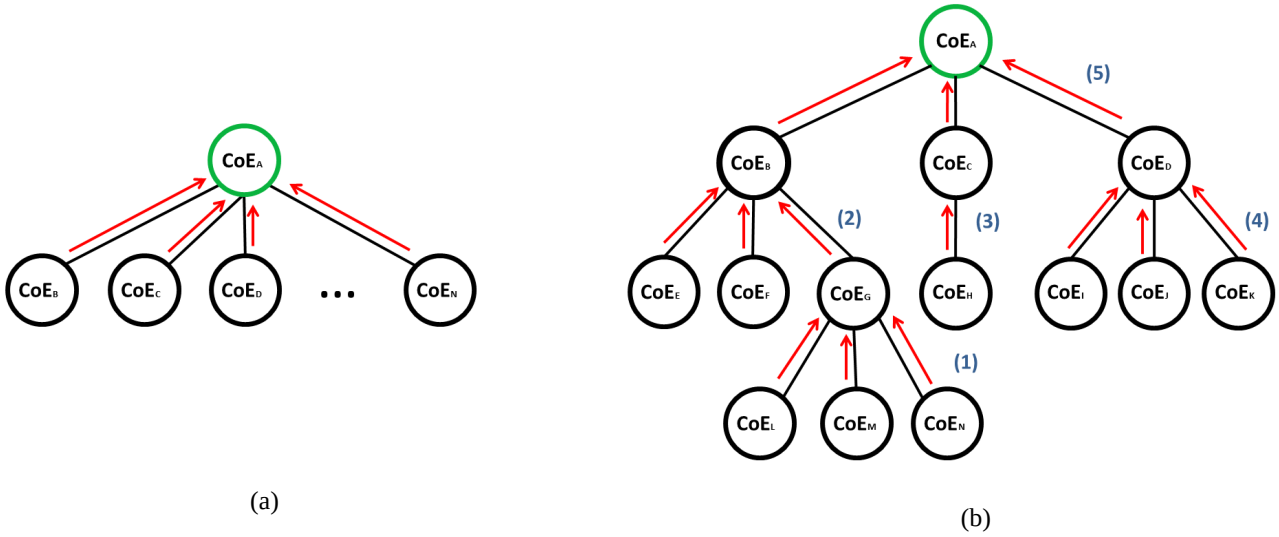

Figure 2**:** Evaluation Use Cases derived in respect to SECCRIT [4] case studies. The subfigure (a) illustrates the basic model of a general tree where depth of the tree is one and the degree is N. This is an initial model where the algorithm introduced in Figure 3 calculates iteratively the conjunction of SPV bits to determine the overall assurance of the leaves of ACS for CoE_i, $i \in$ {B,C,D...N} and aggregates towards root according the policies defined in Table 2 and equasions 8-12. Although this is straight forward, in subfigure (b), the same process is aligned with the post-order traversal method, which at the end aggregates the Assurance towards the root COE_A.

appropriate DAL_{VS}. The DAL_{VS} is not only used for mapping the calculated ACS_{AL}, but also to customize the underlying security properties of a particular AL. Finally, we calculate the AL of the root CoE for a particular ACS, equation 9, depending on the SPs that the CoE corresponds to the AL of the ACS, where by the equations 10 and 11 have to be fulfilled. However, in case when we deal with multiple ACs per CoE we have to consider the equation 12 where we consider the AL of individual AC to determine consolidated AL for a CoE.

Table 3**:** Assurance Level per distinct Assurance classes depicted with Hexadecimal vectors. We define minimal assurance level requirements (DAL_{VS}) of the objects that are in direct relationship with the parent object. It also defines the assurance level requirements per each level of the parent object itself, AL_{VS}. Additionally we define the minimum requirement for each AC in terms of AL, i.e. we define at which assurance level individual AC has to satisfy to define the overall assurance of the object. In case when we have multiple AC to consider in order to derive the overall AL we use the Consolidated Assurance Level (CAL).

ASSURANCE LEVEL		I	II	III	IV	V	VI	VII	N
AC_1	AL_{VS}	0002	0008	0010	0080	0C00	7000	8000	8000
	DAL_{VS}	0002	0004	0030	00C0	0D00	3000	C000	8000
	CAL	-	AL_1	AL_2	AL_3	AL_5	AL_6	AL_7	AL_N
AC_2	AL_{VS}	0002	0008	0020	0040	0300	1800	4000	8000
	DAL_{VS}	0004	0018	0020	00C0	0300	1C00	6000	8000
	CAL	-	AL_1	AL_3	AL_4	AL_5	AL_6	AL_7	AL_N
AC_3	AL_{VS}	0002	000C	0010	00C0	0200	0C00	4000	8000
	DAL_{VS}	0006	0004	0030	01C0	0200	1C00	6000	8000
	CAL	AL_1	AL_2	AL_3	AL_4	AL_5	AL_5	AL_6	AL_N
AC_N	AL_{VS}	0006	000C	0030	00C0	0D00	1C00	7000	8000
	DAL_{VS}	0006	000C	0030	00C0	0D00	1C00	7000	8000
	CAL	AL_1	AL_2	AL_3	AL_4	AL_5	AL_5	AL_7	AL_N

D. General Tree Model

A general tree G is a finite compound set of nodes such that there is only one designated node R, referred as root of the tree G, where each individual node has only one ancestor (Parent) node, with exception of the root, and multiple successors (Children). Each node of the tree is defined with two properties: Depth and Degree. Depth of the node is the distance of the node from the root node, and Degree of the node is the number of successors for a particular node. Moreover, each general tree can be partitioned in n > 0 disjoint subsets T_0, T_1, T_2 … T_{n-1}, where each of which is a tree whose roots R_0, R_1, R_2 … R_{n-1} are children of the tree G. The subset T*i* ($0 \leq i \geq n$) we refer as a subset of trees of T.

Although we intent to depict our services through a general based tree model, they can be also depicted via the binary tree model, since the general tree model is easily transformed to binary tree. For demonstrational purpose of our algorithm (Figure 3) we will use the general tree model. Since the model can be easily transformed, our implementation can be adopted to apply the algorithm on binary trees as well. However, we won't address the assessment of binary trees as it exceeds the scope of this work.

IV. EVALUATION

For evaluating our approach and explaining them in more details, a scenarios are given below in which our approach is applied. As a first step, we need to understand the cyber-risks that exist in the use case scenario and then address the appropriate security properties that need to be assessed and certified.

Perceptions of risk in the context of cloud computing should be well understood, since they will inevitably influence decisions about cloud adoption or the security controls that will be applied to them. Two important factors, amongst others, that must be taken under consideration for better understanding of cyber-security risks, are: (i) the threats and their likelihood

to occur; and (ii) the vulnerabilities and an indication of their severity. A key challenge when understanding the risks associated with cloud computing is to determine those that are specific to the use of cloud.

Therefore, in order to comprehend cloud-specific risks of our scenario, we used the cloud vulnerability catalogue that SECCRIT project [4] has developed, in which we then mapped the Notorious Nine Top threats from CSA [1]. Further on, with the help of CUMULUS project's [2] security property catalogue, we mapped these vulnerabilities to possible security properties for their assessment. The basis of this catalogue is the identification of a number of categories that enable us to focus directly on cloud-related issues. The core of these categories is based on the NIST essential cloud computing characteristics [3].

```
begin procedure:
  for i=k … i=1 do
    if (∀ CoE_C (SPV[i]) ∃! AL_M, M ∈ {1,2,…,7}) {
        AL = M;
        end procedure
    }
    else if (∏_{i=1}^{i=n} CoE_i(SPV[i]) == 0) {
       discard ∀ SPV where SPV[i] =1;
       continue;
  }
    else (∏_{i=1}^{i=n} CoE_i(SPV[i]) == 1) {
       discard ∀ SPV where SPV[i] =0;
       continue;
    }
end procedure
```

Figure 3: *Assurance level calculation procedure* (ALCP) for associated objects used in equation 8. The procedure does the bitwise conjunction of the most significant bit and based on the result decides whether to discard the SPV that have 0 or 1 assigned to a particular bit that is being analysed. Furthermore, during each iteration, the procedure checks if the remaining vectors that define particular component are subset of one of the vector sets associated to a particular AL_i, shown in Table 4, for a particular AC_k

A. *Use Caseses*

The aim of the evaluation is to illustrate the real world scenario via the abstraction of a general tree model, which is used to assure the public safety of critical infrastructure services and assess the assurance according to a set of security classes/properties. We refer in particular to the case studies from the SECCRIT project [4] in order to abstract our approach and make a proof of concept assessment algorithm.

For demonstrating our algorithm we abstract a service via the use case scenarios explained below. Moreover, we implemented our assurance algorithm in Java so we can randomly define properties of evaluation such as depth of a tree, degree of a node, security property vector bit length. Furthermore, found our implementation on post-order tree traversal model in order to efficiently evaluate the assurance of service as a whole.

For the first use case scenario, the Depth (D_1) of the tree T_1 is 1, meaning that we have only a root with set of children and Degree (D_2) will be N which is going to be generated randomly, as shown in (Figure 2 (a)). In the second use case both degree and depth properties are predefined, e.g D_1=3 and D_2=3 (Figure 2 (b)). With the second use case we want to demonstrate appliance of our algorithm on a more complex general tree, which would illustrate the service more realistically.

B. *Security Properties, Vulnerabilities and Threats*

If we refer to the SECCRIT case studies the risk is mostly about authorisation of users, data storage and data leakage. In Figure 4 we present the architecture of the system with components in different levels, as well as their dependencies. Moreover, some relevant security properties are mapped in each component that need to be certified in order to assure the whole service.

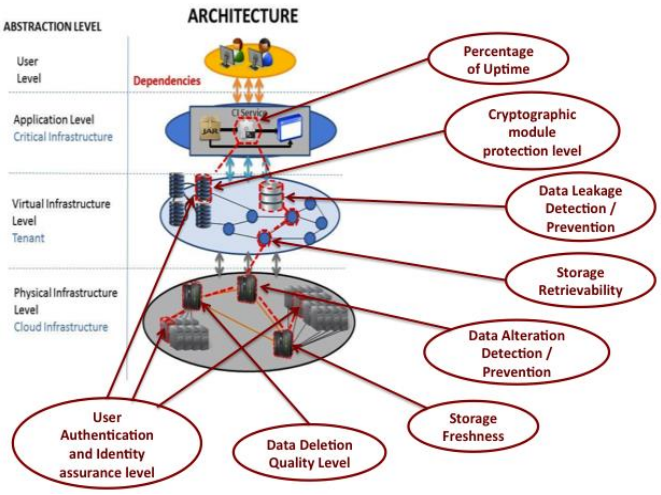

Figure 4: Identified set of Security Properties across various architecture layers of the cloud environment, mapped towards the SECCRIT vulnerability catalogue and CUMULUS property catalogue. Due to the fact that both catalogues enumerate large number of properties we only illustrated most representative ones for time being and will provide more detailed catalogue in our further work.

Table 3 below presents some security properties that are relevant for the case study, their security property category, as well as the vulnerabilities and threats that are related to each one of them. Moreover, the dependencies between these properties are also provided, according to Figure 4. From this list we have selected four properties, e.g. SP_7, SP_4, SP_6 and SP_1 to proceed to the evaluation of our approach, as a starting point of our on-going research on multi-layer assurance dependencies policies.

C. *Scenario based assessment*

For demonstration of our approach we distinguish two specific use cases that are going to be evaluated: Fundamental general tree model (illustrated in Figure 2- a) and advanced tree model (illustrated in Figure 2- b). Both models illustrate a service through a general tree model, where each individual node represents a standalone entity of the particular service that is being evaluated. Furthermore, we use our set of identified security properties to demonstrate our approach by

distinguishing four most relevant properties SP_7, SP_4, SP_6 and SP_1 assigning them as SP_4, SP_3, SP_2 and SP_1 respectively. We implemented a random bit vector generator that generates four bit sets, regardless of the use case, and associates them with individual SPV for a particular object.

For the evaluation of our first use case scenario we fundamentally illustrate a general tree model for each COE_i, $i \in \{B, C, D \ldots N\}$ generated SPV [SP_4, SP_3, SP_2, SP_1], as shown in Table 5 (a). We use the traversal post order method to recursively assess the use case scenarios.

Table 5: Security Properties, Vulnerabilities & Threats

ID	Security Property	Category	Vulnerability	Threats	Dependencies
SP_1	User Authentication and Identity assurance level	Identity Assurance	Loss of human-operated control point to verify security and privacy settings	Data Breaches Data Loss Shared Technology Vulnerabilities	None
			Insufficient authentication security, e.g., weak authentication mechanisms, on the cloud management interface	Account or Service Traffic Hijacking Insecure Interfaces and APIs Malicious Insiders	
SP_2	Data deletion quality level	Data Disposal	Data recovery vulnerabilities, e.g., unauthorised access to data in memory or on disk from previous users	Data Breaches Account or Service Traffic Hijacking Insecure Interfaces and APIs Malicious Insiders Insufficient Due Diligence	None
SP_3	Storage Freshness	Durability			
SP_4	Data alteration prevention / detection	Integrity	Poor/ no integrity checks of the billing information	Data Breaches Insecure Interfaces and APIs Insufficient Due Diligence	SP_1, SP_2, SP_3
SP_5	Storage Retrievability	Durability	Poor/ no backup & restore strategy is in place to prevent the loss of billing information, e.g., in the case of a system failure	Data Breaches Insecure Interfaces and APIs Insufficient Due Diligence	SP_4
SP_6	Data leakage detection / prevention	Data Leakage	Poor/ no encryption of the VM data through a wide-area migration process	Data Breaches Malicious Insiders Shared Technology Vulnerabilities	SP_5
SP_7	Cryptographic module protection level	Key Management	Unmonitored and unencrypted network traffic between VMs is possible, e.g., for VMs on the same node through virtual network Unencrypted physical storage, which is the underlying for allocated virtual storage of the VMs	Insufficient Due Diligence Shared Technology Vulnerabilities Data Breaches Malicious Insiders	None
SP_8	Percentage of Up Time	Availability	Poor/ no implemented QoS (Quality of Service) services, e.g., to guarantee connection bandwidth required by the cloud user Only one ISP connection is considered for operation	Insufficient Due Diligence Shared Technology Vulnerabilities	SP_6
			Poor/ no failover mechanism, e.g., in case of losing one out of two ISP connections Missing redundant power connection leads to a higher risk of losing power	Denial of Service	

Due to the simplicity of the first use case scenario the traversal post order method only determines the sequence of evaluated objects, which is {B, C, D … N} since we have one-level deep tree. In consolidation with our procedural algorithm from Figure 3, we conduct bitwise conjunction. In particular, we start by conducting the procedure illustrated in equation 8 and implemented in our algorithm in Figure 3 on the SP_4. The result of this was 0, which indicates that according to Table 1 we discard all potential combination that fulfil SP_4 (upper eight combination 8-15) and reduce to 3-bit vector set for further evaluation. Our next sequential step, applies the same process on SP_3 resulting also to 0, which also lead to the same outcome, but reducing it into 2-bit vector. The next iteration for the SP_3 resulted to 1 that maps the remaining bit vector sets towards the assurance level two, therefore making the last bit irrelevant for the assurance since both potential outcomes (0 and 1) would lead towards assurance level two. Hence, the final vector, according to Table 1, associates the underlying Associated component set (ACS) of the root node with AL=2 is SPV = [001X]. This process is derived for each AC until we derive final SPVi for each ACi. The final aggregation towards the root is defined with equation 8 which leverages the policies of Table 6 to decide whether both conditions of DAL_{VS} and AL_{VS} are satisfied to determine root assurance level(the equations 9, 10 and 11 have to be fulfilled.), which in this particular case is $CoE_A(AL)=2$. However in case when we deal with multiple AC then we additional have to check weather for each AC satisfies minimum CAL to fulfil a particular AL, stated in Table 2 and defined by equation 12.

Table 7: Randomly generated SPV per individual CoE for demonstrating our algorithm Figure 3, via the use cases from Figure 2. Left table (a) is referring to the first use case scenario, Figure 2 (a), and table (b) refers to the second use case scenario Figure 2 (b).

(a)

	SP_4	SP_3	SP_2	SP_1
CoE_A	0	1	1	1
CoE_B	0	1	1	0
CoE_C	0	0	1	0
CoE_D	0	0	1	1
CoE_N	1	0	1	0

(b)

	SP_4	SP_3	SP_2	SP_1
CoE_A	0	1	1	0
CoE_B	1	0	0	0
CoE_C	1	0	1	0
CoE_D	1	0	1	1
CoE_E	0	1	0	1
CoE_F	0	1	1	0
CoE_G	1	0	0	1
CoE_H	0	1	1	0
CoE_I	1	0	0	0
CoE_J	1	0	0	0
CoE_K	1	0	1	1
CoE_L	0	1	0	0
CoE_M	0	1	1	1
CoE_N	0	1	0	1

In order to evaluate the second use case, advanced tree model (illustrated in Figure 2- b), we generated for each COE_i, $i \in \{A, B, C, D \ldots N\}$ SPVi Table 5 (b). Due to the fact that the first use case tree is a subset of the tree in the second use case, we can apply iteratively the whole process conducted in the first use case scenario, until we aggregate the assurance towards the root. Therefore in order to avoid redundancy we will just refer to the process that we already explained in the first use case and extend it accordingly. The traversal post order method in the second use case, Figure 2(b), has the following sequence {D, F, L, M, N, G, B, H, C, I, J, K, D, A}. Therefore we marked 5 steps in Figure 2(b) that illustrate this

procedure. The first step will aggregate the assurance for ACS_G = (COE_i, $i \in$ {L, M, N}) with the ALCP procedure which outcomes $CoE_G(AL)$ = [010X]. Then, as the second step, when we have the Assurance level of CoE_G we aggregate the assurance of ACS_B = (COE_i, $i \in$ {E, F, G}), e.g. $CoE_B(AL)$ = [0110]. The third step determines the assurance level of CoE_C directly according to one child node CoE_H, $CoE_H(AL)$=[0110]. The fourth step aggregates the assurance level of ACS_D = (COE_i, $i \in$ {I, J, K}), $CoE_D(AL)$=[1001]. Finally the last step of the assessment process is to aggregate the assurance level of ToE_{AL} = ACS_A = (COE_i, $i \in$ {B, C, D}), where $CoE_D(AL)$=[0110], by fulfilling the equations 9, 10 and 11 which lead to the overall assurance of AL=4. However, just as like in the first use case scenario, if dealt with multiple assurance class we have to use equation 12 to derive the final consolidated AL for a particular CoE.

V. Conclusion and Future Work

Within this paper a model and methodology for the assurance of cloud properties across all cloud layers (including system, networking and management aspects) has been introduced. The model specifically has been devised for multi-tenant environments and helps to determine the assurance level of cloud services. The scheme has been evaluate using scenarios. Our evaluation shows efficient application of our proposed assurance assessment model over the two use case scenarios where we demonstrated how services could be assessed according to a set of security properties with defined set of policies.

Based on this work the next steps will provide a complete assurance class and security property catalogue that comprehensively covers the different aspects of cloud environments. Furthermore, we are planning to use real-world applications within the SECCRIT and CUMULUS projects and benchmark them using the introduced scheme. As far as the model itself is concerned we will also further investigate the use of a binary tree model instead of the currently used general tree model, since we can easily transform general tree to a binary tree model, in order to empirically evaluate the performance of our algorithm.

Acknowledgment

The research presented in this paper has been funded by the European Commission, in the context of two Seventh Framework Programme (FP7) projects, the SECCRIT (Grant No. 312758) and the CUMULUS (Grant No. 318580).

References

[1] CSA, “The notorious nine: Cloud computing top threats in 2013”, v.1.0, Cloud Security Alliance, February 2013, available from: http://cloudsecurityalliance.org/research/top-threats/ [retrieved: April 2014]

[2] CUMULUS project, http://www.cumulus-project.eu/

[3] P. Mell and T. Grance: The NIST Definition of Cloud Computing. Technical Report Special Publication 800-145, National Institute of Standards and Technology (NIST), September 2011.

[4] SECCRIT project, https://seccrit.eu/

[5] CUMULUS Deliverable “D2.1 Security-aware SLA specification language and cloud security dependency model v1.01”, September 2013. Available from http://www.cumulus-project.eu/.

[6] Common Criteria (CC) for Information Technology Security Evaluation, CCDB USB Working Group, 2012, part 1-3. [Online]. Available: http://www.commoncriteriaportal.org.

[7] Scholler, M., Bless, R., Pallas, F., Horneber, J., & Smith, P. (2013, December) “An Architectural Model for Deploying Critical Infrastructure Services in the Cloud“, Cloud Computing Technology and Science (CloudCom), 2013 IEEE 5th International Conference on (Vol. 1, pp. 458-466). IEEE.

[8] Florian, M., Paudel, S., & Tauber, M. (2013, December), “Trustworthy evidence gathering mechanism for multilayer cloud compliance”, Internet Technology and Secured Transactions (ICITST), 2013 8th International Conference for (pp. 529-530). IEEE.

[9] S. Paudel, Tauber, M., and Brandic, I., “Towards Taxonomy based Software Security Standard and Tool Selection for Critical Infrastructure IT in the Cloud”, The 8th International Conference for Internet Technology and Secured Transactions (ICITST-2013), 2013.

[10] A. Hudic, T. Hecht, M. Tauber, A. Mauthe and S. E. Cáceres, "Towards Continuous Cloud Service Assurance for Critical Infrastructure IT", The 2nd International Conference on Future Internet of Things and Cloud (FiCloud-2014), 2014

[11] M. Krotsiani, G. Spanoudakis, and K. Mahbub, “Incremental certification of cloud services,” in SECURWARE 2013, The Seventh International Conference on Emerging Security Information, Systems and Technologies, 2013, p. 7280.

[12] Cloud Security Alliance, Cloud Controls Matrix, Available from: https://cloudsecurityalliance.org/research/ccm/

[13] Payment Card Industry Data Security Standard (PCI DSS) Cloud Computing Guidelines, Available from: https://www.pcisecuritystandards.org/security_standards/documents.php?document=dss_cloud_computing_guidelines

[14] COBIT, IT Assurance Guide: Using COBIT, Control Objectives for Information and related Technology, 2007, information Systems Audit and Control Association.

[15] IT Baseline Protection Catalogs, Available from: http://www.bsi.de/gshb/index.htm

[16] National Institute of Standards and Technology, "Information Security Handbook: A Guide for Managers," NIST Special Publication 800-100, October 2006.

[17] ITAF, Information Technology Assurance Framework, 2nd ed., Information Systems Audit and Control Association, 2013.

[18] ENISA, Cloud Computing Information Assurance Framework, 1st ed., European Union Agency for Network and Information Security, 2009. Available from:http://www.enisa.europa.eu/

[19] Top Threats Working Group. "The Notorious Nine: Cloud Computing Top Threats in 2013." Cloud Security Alliance (2013).

[20] Jerry Busby, Lucie Langer, Marcus Schöller, Noor Shirazi and Paul Smith Deliverable: 3.1: "Methodology for Risk Assessment and Management", 2013, Available online: https://www.seccrit.eu/upload/D3-1-Methodology-for-Risk-Assessment-and-Management.pdf

[21] Patel, Pankesh, Ajith H. Ranabahu, and Amit P. Sheth. "Service level agreement in cloud computing." (2009).

[22] Buyya, Rajkumar, Chee Shin Yeo, and Srikumar Venugopal. "Market-oriented cloud computing: Vision, hype, and reality for delivering it services as computing utilities." High Performance Computing and Communications, 2008. HPCC'08. 10th IEEE International Conference on. Ieee, 2008.

[23] OpenStack, http://www.openstack.org/

[24] CloudStack, http://cloudstack.apache.org/

Towards a Unified Secure Cloud Service Development and Deployment Life-cycle

The paper "Towards a Unified Secure Cloud Service Development and Deployment Life-cycle" was published at the the 3rd International Workshop on Software Assurance held in cojunction with the 11th IEEE International Conference Availability, Reliability and Security (ARES) 2016 in Salzburg.

The paper can be found online under the following link: `http://ieeexplore.ieee.org/document/7784602/`

Towards a Unified Secure Cloud Service Development and Deployment Life-cycle

Aleksandar Hudic, Philipp M. Radl, Thomas Lorünser
Austrian Institute of technology
Digital Safety and Security
Vienna, Austria
Email: firstname.lastname@ait.ac.at

Matthias Flittner, Roland Bless
Karlsruhe Institute of Technology (KIT)
Institute of Telematics
Karlsruhe, Germany
Email: firstname.lastname@kit.edu

***Abstract*—Designing and developing cloud services is a challenging task that includes requirements engineering, secure service deployment, maintenance, assurance that proper actions have been taken to support security and, in addition, considering legal aspects. This is unfortunately not possible by taking current methods and techniques into consideration. Therefore, we require a systematic and comprehensive approach for building such services that starts the integration of security concerns from early stages of design and development, and continuous to refines and integrate them in the deployment phase. In this paper we therefore propose a solution that integrates security requirements engineering and continuous refinement in a comprehensive security development and deployment life-cycle for cloud services and applications. Our approach is focused on iterative refinement of the security-based requirements during both software engineering (development phase) and software maintenance (deployment phase).**

***Keywords*-security, cloud, development life-cycle, requirements engineering, security policies, assessment, assurance, monitoring**

I. Introduction

The still ongoing revolution in the usage and consumption of IT resources driven by the cloud paradigm is dramatically changing the ICT landscape. Essentially, the cloud paradigm utilizes the service delivery model to facilitate outsourcing on all possible layers. While some see it as a novel technological concept, others only consider it as an evolutionary step of ICT technologies [1]. Nevertheless, cloud computing plays an important role by providing significant economical and operational advancements, by utilizing interoperability, scalability, on demand service provisioning on global scale with minimum management effort [2].

Prior to being hosted in a cloud, services need to be designed and tailored to maximally leverage the cloud characteristics (on-demand provisioning, ubiquitous network access, resource pooling, rapid elasticity and measured services) [3], by taking design principles and requirements into consideration. A common approach for engineering software products is the *Software Development Life Cycle* (SDLC) [4] also supported by ISO 12207 Standard for software life-cycle processes [5].

The complexity of software, especially in dynamic and volatile environments such as cloud, is hard to predict in early stage of development since it depends on the evolvement and refinement of the initial requirements. During the early stage of software design and development initial high level objectives and requirements that the software needs to fulfill are defined. The requirements are considered the foundation of the software development process very often require to be refined during both development and deployment life-cycle of a product. Therefore, it is necessary to integrate an iterative requirements engineering process that will implement continuous evolvement of both software and requirements during the whole life-cycle of a product [6]. On top of the aforementioned challenges towards designing and developing software products, security is another important requirement nowadays that is often being treated independently of the development process and considered in later stages. Furthermore, its consideration is often premature and security objectives are often traded for usability aspects. However, in our work we emphasize that security engineering should be an integral part of the whole engineering process and carefully considered in each step of a product's life-cycle. In particular, we propose a secure software development life-cycle for cloud services that covers both development and deployment phases of the product life cycle in a cloud environment. Furthermore, our model aligns design, development and deployment with standards, best practices and guidelines as an iterative process that also integrates security requirement engineering and refinement [7], [8].

This work is structured as follows. Section II offers a detailed literature overview of methodologies for secure service development and operation from early design to late deployment and maintenance. Next, in Section III we show the results of a survey that analyzed the relevance of cloud characteristics and design concerns during the development stage. Furthermore, we also analyze and propose a generic security requirements engineering process that we will integrate later in our secure cloud service life-cycle. In Section IV we present our approach for iterative cloud service design and development that integrates security in each step through both development and deployment phase for developing and deploying secure cloud services. Finally, in Section V we conclude our work and outline our future work.

II. State of the Art

Most recent studies indicate that despite the attractive benefits, lack of security (e.g. lack of transparency, data

privacy, trust, data lock-In, data loss) still remains a major obstacle for deploying services into the cloud. The research community is keen on analyzing and addressing security challenges in the cloud environment [9], [10], [11], [12], [13], [14], [15]. This is especially challenging when it comes to critical infrastructure services that require high attention when it comes to security [9], [15]. Nevertheless, there is a part of the research community that proposes solution and methods for service deployment in a cloud. Khajeh et. al [16], [17] proposed, in their work, the migration of enterprise IT services to the cloud with regards to context of financial and socio-technical enterprise issues and the decision making process for service migration by taking into consideration cost modeling and risk assessment. Furthermore, Kaisler and Money, in their work [18], evaluate the compatibility of the service migration approach with the cloud computing paradigm by addressing acquisition, implementation, security, usage reporting, valuation and legislative challenges during the process. In their work, Fehling et. al. [19] elaborate and advise best practices for addressing web based service migration challenges with regards to migration patterns.

Requirements engineering plays an important role in secure service development, because it identifies crucial security considerations that have to be taken into account during the early stage of development till the deployment and maintenance to ensure the complete service lifecycle. Therefore, in their work Haley et.al. [20] present a comprehensive framework for security requirement analysis and elicitation based on context analysis. Security requirements are defined as constraints on system's functional requirements based upon system or service security goals. Furthermore, Mellado et.al. [21], [22] leverage the Common Criteria approach for to utilizing requirements engineering with respect to security concerns in early development stage, formally referred as Security Requirements Engineering Process (SREP). The SREP is an asset-based and risk-driven model that elicitates security requirements through iterative micro-processes (e.g. identifying, prioritization and categorizing requirements, vulnerabilities and threats, assessing risks, and identifying security objectives). The work of Hesse et.al. [23] outlines an approach that combines heuristics, monitoring and decision documentation to perform semiautomatic security requirements engineering, whereby heuristics monitoring is used to mitigate the manual effort.

Maintaining and ensuring that security of our systems and services are at the proper level we require solutions (e.g., security assessment, security monitoring, auditing) that will perform the validation or assessment of security in our ICT systems. One of the most prominent approaches nowadays, Common Criteria [24] provides an efficient and systematic approach for security assessment that offers a certain level of confidence that predefined set of security requirements (i.e. functional or assurance requirements) have been met by the evaluated product formally refereed as Target of Evaluation(ToE). The pioneering work of security assessment in ICT environments was developed by the US Department of Defense under the name Trusted Computer System Evaluation Criteria (TCSEC) [25], commonly referred to as the "The Orange Book". The approach classifies the assessment across three fundamental categories minimal, discretionary, mandatory and validated protection where each category contains a proposed set of security controls that are being validated (e.g. policies, access control models, audit trails, roles, processes, etc.). Furthermore, TCSEC is unfortunately mostly focusing on confidentiality and towards a high-level evaluation of systems and services. Analogous to the TCSEC approach, the Information Technology Security Evaluation Criteria (ITSEC) [26] also builds the evaluation on security controls. However, the users are able to encounter the preferred set of security requirements tailored to their product which the ITSEC formally refers to as Target of Evaluation (ToE). In addition, the ITSEC while still assigning levels it differentiates between functional and assurance levels.

With respect to the above state of the art our work overcomes the gap of a holistic integration of security engineering during both development and deployment phase, i.e., it offers continuous security integration that is based upon iterative security requirement engineering. Our security requirements engineering process provides security requirements in various abstract forms (objectives, requirements and properties) to support secure service design, monitoring and assessment through development and deployment phases. Our solution present an uniform solution that is aligned with the approaches proposed by Wagner et.al. [27] and Hudic et.al. [28].

III. Information Security Requirements Analysis and Engineering

Engineering and elicitation of requirements for supporting software design and development is a cumbersome and time consuming process, especially when it comes to security. Often, due to the very strict deadlines that are dictated by time to market, security is not considered as a primary concern in service design or development. Such products are easily being prone to security flaws because of some minor mistake during design stage [29]. Hence, design and development of secure services becomes additionally challenging when it comes to deploying those services in cloud environments, because there are many challenges entailed with the features offered by the cloud that have to be carefully taken into consideration for both development and production phases [30].

Design, followed by development, of a service life cycle commonly starts with defining very high level objectives that have to be engineered into the development process of a service and maintained until a service is finally deployed. To support such analogy it is often necessary that through an iterative refinement process security requirements are continuously improved before being integrated in a particular stage. By taking these into consideration, we propose and develop an approach for continuous integration, refinement and maintenance of security related requirements

during both development and deployment phase. Prior to building our secure cloud service life cycle we analyze relevance and depict a conceptual model for the security requirements engineering process for cloud applications and services, shown in Figure 3.

A. Requirements Analysis

In order to support our requirement engineering model we conducted a survey among 31 academic and 46 industry professionals where we investigated the relevance of cloud characteristics and design concerns during the development phase. The first objective of our survey was to investigate if the participants take under consideration, in early stage of design and development, the following criteria: risk assessment, SLA management, architectural patterns, service life-cycle, autonomic security management, forensics and auditing, and international standards. The results, depicted in Figure 1, show that generally industry participants have higher interests in the above mentioned objectives than the academic participants with the exception of autonomic security management that was slightly below (80%) for industry participants. Although generally lower than industry, the academic participants focus their attention towards risk assessment (82%), forensics and auditing (75%), international standards (67%) and autonomic security management (75%).

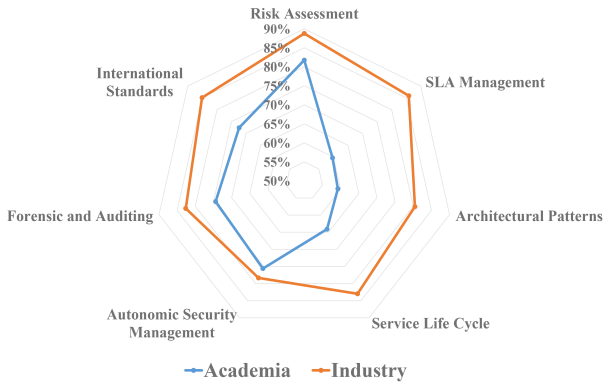

Figure 1. Relevant design concerns

The second objective in our survey is focused to investigate how the NIST cloud computing characteristics [3] are embraced when designing, developing and deploying services. The results, depicted in Figure 2, show high interest for both industry and academic participants for considering cloud characteristics when designing, developing and deploying their services. The results, shown in Figure 2, indicate also in case of NIST cloud characteristics that generally interest by industry participants is higher than for academic participants. However, the geolocation was the only characteristic where the industry had a slightly lower interest than the academic participant, and at the same time characteristic with the lowest interest results for industry participants. This is most probably due to the higher interest in industry for usability over security when it comes to developing products.

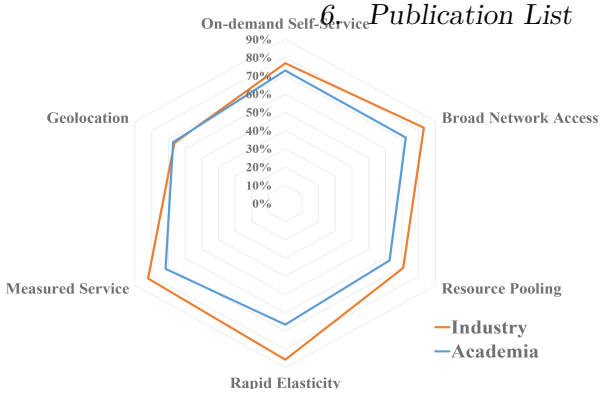

Figure 2. Relevance of NIST cloud characteristics during service design and development phase

B. Requirements Engineering Model

We envisioned the security related requirements engineering and refinement, shown in Figure 3, as multi stage iteration process that starts with the initial context analysis for a particular use case scenario, i.e., application or service being developed or deployed in cloud environment. The output of the context analysis are high level business objectives, functional requirements, and security objectives that are aligned with standards, guidelines and best practices. Next step in our process performs an analysis and refinement of the high level requirements and objectives based upon risk and vulnerability assessment. Hence, the output of service risk and vulnerability assessment is taken in to consideration when performing refinement of security objectives. The result of security objective refinement are concise security requirements that are used for secure service design, development and deployment. After the service is deployed in production, validating security during runtime is supported by the refined version of security requirements that are defined as security validation elements, i.e., security properties. Both security requirements and properties must be aligned with standards, guidelines and best practices. Furthermore, both security properties and security requirements are used for defining security related policies for maintaining, auditing, designing and assessing services during both development and production phases.

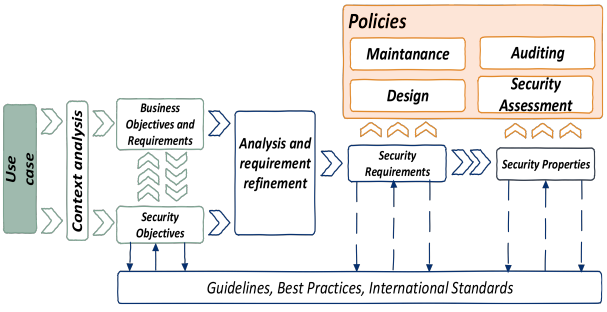

Figure 3. Iterative process for building security policies and security requirements engineering

IV. Secure Cloud Service Life-cycle

A software service life-cycle comprises several stages ranging from design, development, deployment to mainte-

nance. To improve the overall software quality and ensure its security it is crucial to consider security aspects in all stages. Commonly, service design starts with high level business and security requirements. They are afterwards translated to functional and non-functional requirements and used throughout the development phase to implement software functionalities. Whenever a particular service is being deployed or migrated to an environment such as cloud we need to define additional requirements together with a deployment strategy [27]. Furthermore, we would also like to have a guarantee that the security requirements are fulfilled at design, implementation and maintenance of a cloud service. In order to check compliance to defined security requirements, measurable assessment criteria have to be defined which can be used in validation process. In summary, we see that the protection of software through its entire life-cycle requires elicitation and refinement of security requirements in all stages which can become challenging when targeting heterogeneous and distributed infrastructures. In the case of cloud usage a new abstraction level has to be applied to support mapping between different stages into an unified approach.

We propose a two-phase secure cloud service life-cycle that integrates requirements engineering and iterative refinement with respect to security, through each stage of both phases. The first phase is called *Development phase* and covers the sequential set of steps where a service is being designed and developed. Secondly, the *Production phase* is where a deployed service is validated against those security requirements that have been defined in development phase.

As mentioned before, consistent integration of security concerns throughout each step of both phases is vital for designing and operating secure systems and services. Therefore, in each phase of our proposed life-cycle we conduct an iterative security requirements engineering process to align the requirements to the needs of a particular step (design, development, maintenance, assessment or monitoring) and standards, best practices, or guidelines [31], [32]. Brining together the cloud secure development life-cycle [27] with the cloud assurance assessment framework [28] and cloud inspector [33] offers a unique and smooth way to continuously integrate security in service life-cycle from development phase to production phase. We perform continuous refinement of security in terms of properties that through the life-cycle yield bot security functional and non-functional requirements depending on the phase corresponding step.

A. Cloud Service Development

Security by design approach is a vital part for designing secure software and preparing it for deployment in security demanding environments such as cloud. We introduce an enhanced version of Secure Cloud Service Development Life-cycle model from Wagner et.at. [27], that supports our model for integrating security across all stages of the cloud service life-cycle from design, development, deployment preparation and migration, till production.

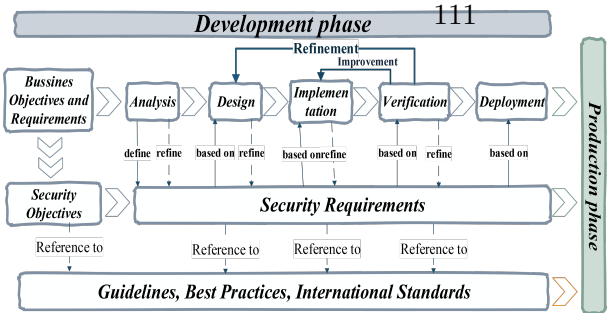

Figure 4. Development phase of secure cloud-service development life-cycle

In our approach we narrow down the focus of the secure development life-cycle process by taking the following objectives into consideration:

- Integration, engineering and continuous refinement of security requirements in each stage of software (design, development, testing, deployment and maintenance) for cloud based architectures.
- Secure software development for a cloud environments from scratch.
- Software migration from a legacy system to the cloud (adoption for cloud).
- Software migration from private to public clouds and vice versa.
- Iterative security requirements engineering during both development and production life-cycle phases.

As mentioned before, we align our approach with Wagner et.al. [27] that is focused on building a guideline for secure service migration in cloud by integrating the security engineering process into the cloud secure development life-cycle. We extend the approach by adding an additional initial step to addresses high level security requirements according to the high level business objectives supported by the continuous security requirement refinements according to standards, guidelines and best practices. Furthermore, our approach highlights the following six stages for the secure software development phase:

a) High level security objectives analysis: This preliminary step consolidates high level business objectives with security related standards, best practices and guidelines to set the initial security objectives for secure service design, development and deployment.

b) Analysis: This step performs analysis of a service with respect to cloud requirements. The IT services that are intended to be developed and cloudified, i.e. hosted in the cloud, are analyzed to prove their eligibility of deployment in cloud environments. Furthermore, the initial set of security requirements is specified and potential threats to the particular use case are identified. Ideally, if security requirements for the IT service are predefined, they are taken into account and, if needed, adjusted to the circumstances occurring in each subsequent stage. In particular, this also involves security requirements indirectly resulting from the cloudification or development of a particular service. For example, this might refer to requirements for providing credible digital evidence on the providers security-related conduct for the potential

case of legal conflicts, or to cloud-specific requirements from data protection law. Moreover, this step also proposes an analysis of the implications that the development or cloudification of the IT service has on the organization and the business.

c) Design: In the design step, the software architecture for the to be migrated or developed IT service is designed in line with the security requirements specified in the analysis step. If required, refinements of the security requirements are performed for precisely aligning the security requirements towards the particular use case.

d) Implementation: The foundations defined in design step are used to implement part or complete service in line with the NIST cloud characteristic. Here we can also take into consideration the NIST cloud characteristic analysis performed in Section III-A. Also in this step additional security property refinements can be performed if required.

e) Verification: In this step, software is tested against the predefined set of security requirements before being deployed or migrated in to the cloud. In addition, in case of cloudifying IT service the readiness of the organization shall be verified (e.g. special disaster recovery strategies, trainings, or revisions of SLAs might be required). If the verification does not succeed, either further implementation effort needs to be taken and/or the design needs to be revised. Additionally, risk assessment is performed in order project the risks involved based upon potential threats and vulnerabilities at this particular stage.

f) Deployment: In the final step of development phase the IT service is deployed to the cloud environment by taking into account the security requirements related to platform configuration.

The result of the development phase is a service that integrates best practices with respect to security aligned with most prominent standards and guidelines nowadays. Furthermore, an early stage security requirements engineering yields from security objectives a more concise set of security requirements, that iteratively through the above mentioned steps lead to more robust and secure design of our services. These requirements are also used to perform the selection of the most eligible cloud provider that can fulfill the needs of a particular customer to host his service and as the input for the production phase where the security of a particular service should be maintained with respect to security requirements.

B. Cloud Service Deployment

The integration of the development life-cycle, that we introduced in Section IV-A, with the assurance assessment framework and CloudInspector offers a unique way to verify the implementation of the initial security objectives from the early stage of development to the production stage. At the same time, these properties are being refined iteratively through stages to meet security related best practices, standards and guidelines. The second phase of our secure cloud-service life-cycle model, the production phase, is focused on maintaining and monitoring security concerns based upon the security requirements defined in

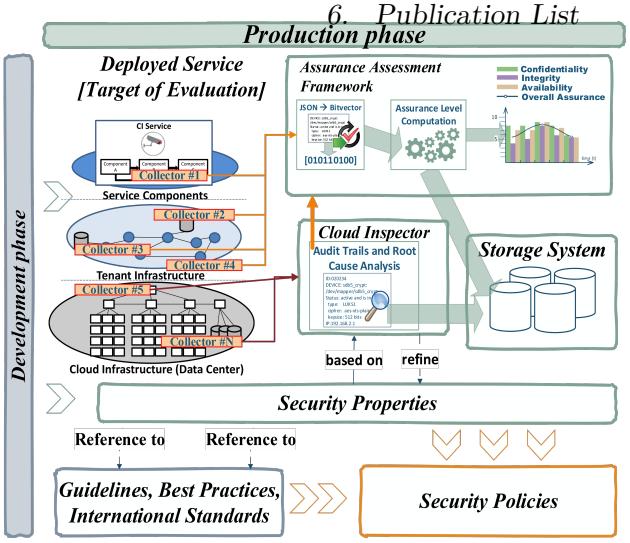

Figure 5. Production phase of secure cloud-service development life-cycle

the development phase.

The development phase, shown Figure 4, enumerates the foundational security requirements and measurable properties of the system that can be used in both assurance assessment framework and CloudInspector, shown in Figure 5. The main difference of those assurance framework and CloudInspector is the service model. CloudInspector is generally envisioned as a module used at infrastructure level to provide on demand audits, whereby assurance framework is covering all cloud levels in a continuous manner. The security requirements have to be aligned with common security guidelines, best practices, and international security related standards. Nevertheless, these security requirements require additional adjustments to be used in the production phase by assurance assessment framework and CloudInspector.

1) Cloud Assurance: The security requirements defined by the development phase are focused to identify functionalities of a service rather than validation and measurement elements. Hence, a requirement engineering process is necessary to transforms development phase security requirements into non-functional production phase security requirement that we refer to as *security properties*. Furthermore, the security properties are then used by the security assurance framework developed by Hudic et. al. [28] to acquire security related information from the infrastructure where the evaluated services are being deployed. Since our focus is to evaluate large complex ICT infrastructures such as cloud the assurance assessment framework is of a particular benefit for our secure cloud service life-cycle. As mentioned, the assurance assessment framework performs security based evaluation of complex multi-layered infrastructures with respect to a specific predefined set of security properties, i.e., quantitative security analysis of a particular entity hosted in the cloud. The foundation of the quantitative security assessment criteria is aligned with the Common Criteria, a comprehensive and

systematic approach developed by Department of Defense for performing security assessment. Furthermore, our assurance security assessment approach uses the following three elements to systematically organize the security assurance assessment of particular system or a service: Target of Evaluation (ToE), Group of Evaluation (GoE) and Component of Evaluation (CoE). These entities are used for creating holistic service abstraction and, at the same time, offer flexibility, granularity and precision for continuously assessing or validating security concerns across a wide area of components, groups of components, and even entire ICT infrastructures. Nevertheless, to perform consistent security assessment framework requires a predefined set of security properties, used to validate security across individual components of interest (CoE). These security properties need to be concisely defined in order to be used by the assurance assessment framework security validation based upon their conditions. Therefore, we use the security requirements from the development phase to engineer the security properties for our validation process. To acquire security related information across the infrastructure where a particular service is deployed we use *Collectors* that according to the precise definitions of individual security property harvest information across the whole infrastructure and deliver it to assurance framework to compute the assurance level. Additionally, both security requirements and properties are used to define policies for maintaining or assessing security, and even guidance for secure service development and design. The collected security related information across the observed infrastructure is then compiled to security levels, which we refer to as assurance levels, and classified across three assurance classes (confidentiality, integrity, availability).

2) Cloud Inspector: As important as monitoring of the assurance level with the security assurance framework is to verify if the cloud provider fulfills contractually agreed security policies (such as geo-location of virtual machines, dedicated host requirements, or physical host anti-affinity) during runtime. The current best practice for that task is certification (e.g., [31], [34], [35]) of cloud provider practices only in large intervals. This does not permit continuous transparency of fulfillment of security policies during runtime. Additionally, in case of data protection, the law states in many countries that a processor of personal data has to be actively controlled. Therefore, we propose to use an independent Transparency-as-a-Service solution such as *CloudInspector* [33] to overcome the lack of transparency in cloud computing.

The *CloudInspector* solution consists of two functionalities: on-demand auditing and continuous evidence gathering. *CloudInspector* enables tenants to continuously control contractual agreements (security policies or properties) during the cloud deployment phase (runtime). Additionally, *CloudInspector* continuously collects meta data about current cloud behavior. In case of a dispute in court the collected meta data at best could be used to determine the root cause of a failure (i.e. negligence of cloud provider). To do so, meta data about cloud behavior must be collected beforehand during deployment phase. The *CloudInspector* solution therefore gathers information within the cloud environment independently of the cloud management platform. Due to the independence the *CloudInspector* solution is able to unveil misconfiguration or malfunction of the cloud management platform. The Transparency-as-a-Service solution consists of two elements: Transparency Controller Module (TCM) and Transparency Enhancement Module (TEM). Whereas the TCM provides an audit request interface for tenants. The TEMs are distributed monitoring agents on each physical host within the cloud environment. The TCM serves as interface for tenants, processes on-demand audit inquiries and coordinates audit requests to as well as responses from the distributed TEMs. The TCM offers per tenant access via web-based or RESTful interfaces. A TCM transforms incoming audit inquiries of tenants into internal audit requests. Audit requests are used for on-demand real-time auditing. These audit requests are sent via audit channels to the TEMs.

After all audit results of the corresponding TEMs are received, the TCM evaluates them and prepares an audit response for the tenant. The TEM monitors physical and virtual resources residing on a physical host and uses data sources that are largely independently of the cloud management platform. Only a single TEM per physical host (compute, network, storage) is necessary. A TEM gathers *audit data* on-demand (i.e., due to an audit request) or continuously according to individual tenant policies. Furthermore, a TEM consists of four basic types of components, namely *Collectors*, *Manager*, *Analysis*, and *Logging*. Collectors are connected to different cloud management platform independent audit sources, e.g., information may be gathered from the hardware, the operating system, event logs, the virtualization library or other physical components. For example, a list of active virtual machines can be obtained by using an operating system command and/or using an API from the hypervisor. Each collector is instructed by the Analysis (i.e., on-demand auditing) or Logging component (i.e., continuous evidence gathering) to gather specific audit data. This can happen regularly by polling certain values or it can be event based so that other components only have to act on such events. The Manager component detects changes of tenant assigned virtual resources (e.g., creation, deletion, migration, start, stop). Depending on those observations the manager joins or leaves an audit channel. The manager component receives audit requests from a TCM via audit channels and sends back audit results directly. If an audit request is received the manager instructs the analysis component to process the audit request (on-demand auditing) or triggers the logging component to continuously record related events (continuous evidence gathering).

The Analysis component processes incoming audit calls from the manager. It either performs a lookup in recently locally recorded data or performs an on-demand check. It may trigger one or several collector components to gather audit data and preprocesses information according

to the specific audit request. Finally, it returns the corresponding audit data to the manager component. The logging component allows tenants to initiate policy-based continuous evidence gathering in order to create audit trails. Based on tenant policies this component may continuously trigger collectors or record certain events to gather relevant audit data. For instance, it could log the operating system, cloud platform and hypervisor versions twice a day or any kind of management access, hardware failures, or reboots at any time when they occur. The usage of *CloudInspector* during the deployment phase guarantees that tenants are able to verify if the cloud provider fulfills contractually agreements during runtime and that in case of a dispute in court significant evidence about cloud behavior will be available.

C. Application scenario

To demonstrate the application of our life-cycle on a real world scenario we highlight major steps of process for building, deploying and maintaining a video surveillance service for critical infrastructure, such as public safety, deployed in a cloud environment. The initial objectives of our video surveillance service is to identify potentially malicious behavior, especially in high security areas. Therefore the video surveillance systems has to be able to process video recordings in real time and identify potentially malicious individuals. Since the prior objective of the video surveillance software is the facial recognition functionality that performs authentication of individuals. The high level requirements of our service are:

- identifying and authorizing individuals,
- providing high availability of video surveillance service without downtime,
- protecting the confidentiality and integrity of video records.

The analysis phase in this case would identify functional security requirements in line with the above objectives: secure service auditing, restricting access to sensitive video recordings, ensuring high availability and protecting confidentiality of video recordings. Further, more detailed, analysis outlines a more concise set of security requirements in line with the ISO 27001/27002 and NIST 800-53 standards that offer a comprehensive list of security requirements with comprehensive description. Additionally, as part of the analysis step we performed a risk assessment based on potential threats that can occur (e.g., malicious insider that could temper or destroy video surveillance records, broken disk containing unencrypted video records being lost, hosting data under legislative domain with invasive privileges to access information, etc). The result of the risk assessment and threat analysis extends security requirements used to develop our secure video surveillance services. In this early stage of design, implementation and verification standards such as ISO 27001/27002 are used to identify best practices for implementing security controls that can ensure confidentiality and integrity of video records. Therefore, during these three process steps we perform additional service refinements and improvements with respect to security requirements.

Furthermore, in the implementation phase of a service we take into consideration the cloud characteristics, depending on the relevance to the use case just as in our analysis in Figure 2, in order to leverage them properly and ensuring high availability of the service when being deployed in cloud. Prior to performing and planning deployment of the cloud service, additional verification and risk assessment analysis is performed to identify unexpected deviations in design or development, and any new potential threats.

Next, we have to consider secure deployment or migration of our service to the cloud by considering the CloudSDLCv1 from Wagner et.al. [27]. Once the service is deployed, we have to further consider measurable metrics to perform the security validation and auditing of the deployed service. In this part we conduct additional security requirement engineering that yields a set of security properties from the security requirements defined in the development phase. They are used for the validation of security goals via the security assurance framework developed by Hudic et. al. [28]. The security assurance assessment framework abstracts the service over individual components to perform independent security validation that is afterwards taken into consideration when performing holistic security analysis, as shown by Figure 6. The right hand side of the Figure 6 shows a general tree model of abstracted video surveillance service where set of security properties is validated across each component. The assurance framework identifies per component, e.g., components 1, 3, 4, 5 and 6, individual security properties that do not fulfill their security control requirement, marked red in right hand side of the Figure 6. The underlying infrastructure information is being delivered by the CloudInspector on demand. This gives us the ability to easily deploy our assurance framework on any given public cloud provider that integrates CloudInspector as independent solution for acquiring infrastructure information. Furthermore, the security assurance assessment framework offers the ability to the end users for defining variety of security validation policies for validating the security concerns of their cloud providers.

In the same way, the CloudInspector solution is used during deployment phase to actively control the cloud provider, to collect evidence for a potential debate in court and to support security assurance assessment framework with infrastructure related information. From data protection perspective recorded video footages of individuals are personal data. If for example a tenant uses this cloud-based video surveillance service to process such personal data of individuals, he has to actively control the cloud provider that he processes the data as ordered (only in datacenters in specific countries or on dedicated physical hosts). Additionally, with CloudInspector actively controls if the cloud provider fulfills contractual agreements. From a civil law perspective continuously collected and stored meta data about cloud behavior is very useful. CloudInspector

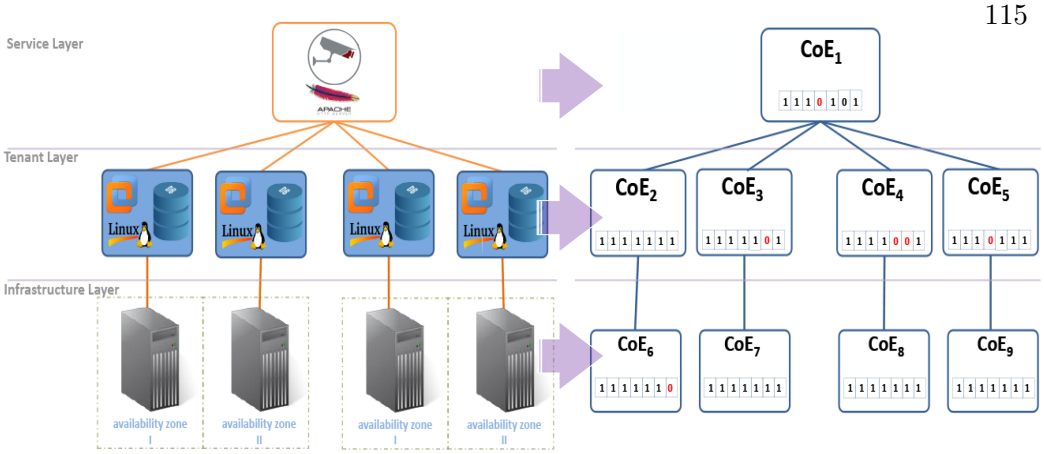

Figure 6. Illustration of holistic abstraction of video surveillance service for security assurance assessment process.

will provide this meta data in case of a dispute in court, so that this meta data can be used during a root cause analysis (which maybe unveil cloud provider negligence).

As shown by our application scenario bringing together secure development life cycle with security assurance assessment framework and CloudInspector enhances the service security by design and increases the transparency with respect to securely hosting services in cloud. If a particular cloud provider integrates the CloudInspector for investigating infrastructure related security concerns that additionally support legal restrictions, we are able to easily deploy assurance assessment framework without interfering with internal cloud processes and exposing internal infrastructure sensitive information and derive security assessment.

V. Conclusion and Future Work

We present in our work a uniform methodology for developing and deploying secure cloud services as a life-cycle that implements continuous integration and refinement of security requirements. Our methodology is built as a uniform and sequential process that integrates design, implementation, testing, deployment, maintenance, assessment and monitoring of cloud services. These sequential steps are divided in two phases, Development phase that covers design, implementation, testing, deployment, and Production phase that covers maintenance, assessment and monitoring. In both phases, each of the mentioned steps integrates security as its essential objective of service evolution. Furthermore, to ensure that security requirements have not only been properly integrated in to a service during design and development but that they are also properly ensured during the deployment, we have integrated security assurance assessment framework and CloudInspector to monitor key security aspects of both deployed service and infrastructure on which the service is being deployed.

Acknowledgments

This work has been supported and partially funded by the European Commission research projects PRISMACLOUD Grant No. 644962 and CREDENTIAL Grant No. 653454.

References

[1] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A view of cloud computing," *Commun. ACM*, vol. 53, no. 4, pp. 50–58, 2010. [Online]. Available: http://doi.acm.org/10.1145/1721654.1721672

[2] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," *Future Generation Comp. Syst.*, vol. 25, no. 6, pp. 599–616, 2009. [Online]. Available: http://dx.doi.org/10.1016/j.future.2008.12.001

[3] P. Mell and T. Grance, "The nist definition of cloud computing," 2011.

[4] I. Jacobson, G. Booch, J. Rumbaugh, J. Rumbaugh, and G. Booch, *The unified software development process*. Addison-Wesley Reading, 1999, vol. 1.

[5] I. O. for Standardization and I. E. Commission, "Systems and software engineering – software life cycle processes," International Organization for Standardization, Standard, July 2008.

[6] P. Forbrig, "Continuous requirements engineering and human-centered agile software development," in *Joint Proceedings of REFSQ-2016 Workshops, Doctoral Symposium, Research Method Track, and Poster Track co-located with the 22nd International Conference on Requirements Engineering: Foundation for Software Quality (REFSQ 2016), Gothenburg, Sweden, March 14, 2016.*, 2016.

[7] C. B. Haley, R. C. Laney, J. D. Moffett, and B. Nuseibeh, "Security requirements engineering: A framework for representation and analysis," *IEEE Trans. Software Eng.*, vol. 34, no. 1, pp. 133–153, 2008. [Online]. Available: http://dx.doi.org/10.1109/TSE.2007.70754

[8] D. Mellado, E. Fernández-Medina, and M. Piattini, "A common criteria based security requirements engineering process for the development of secure information systems," *Computer Standards & Interfaces*, vol. 29, no. 2, pp. 244–253, 2007. [Online]. Available: http://dx.doi.org/10.1016/j.csi.2006.04.002

[9] C. Alcaraz and S. Zeadally, "Critical infrastructure protection: Requirements and challenges for the 21st century," *International Journal of Critical Infrastructure Protection*, no. 0, pp. –, 2014.

[10] S. Subashini and V. Kavitha, "A survey on security issues in service delivery models of cloud computing," *Journal of Network and Computer Applications*, vol. 34, no. 1, pp. 1–11, 2011.

[11] B. Kandukuri, V. Paturi, and A. Rakshit, "Cloud Security Issues," in *Services Computing, 2009. SCC '09. IEEE International Conference on*, Sept 2009, Security, pp. 517–520.

[12] D. Zissis and D. Lekkas, "Addressing cloud computing security issues," *Future Generation Computer Systems*, vol. 28, no. 3, pp. 583–592, 2012.

[13] R. Piggin, "Are industrial control systems ready for the cloud?" *International Journal of Critical Infrastructure Protection*, no. 0, pp. –, 2014.

[14] M. Nanavati, P. Colp, B. Aiello, and A. Warfield, "Cloud Security: A Gathering Storm," *Commun. ACM*, vol. 57, no. 5, pp. 70–79, May 2014. [Online]. Available: http://doi.acm.org/10.1145/2593686

[15] M. Y. A. Younis and K. Kifayat, "Secure cloud computing for critical infrastructure: A survey," *Liverpool John Moores University, United Kingdom, Tech. Rep*, 2013.

[16] A. Khajeh-Hosseini, D. Greenwood, and I. Sommerville, "Cloud Migration: A Case Study of Migrating an Enterprise IT System to IaaS," in *Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on*, July 2010, Service migration, pp. 450–457.

[17] A. Khajeh-Hosseini, I. Sommerville, J. Bogaerts, and P. Teregowda, "Decision Support Tools for Cloud Migration in the Enterprise," in *Cloud Computing (CLOUD), 2011 IEEE International Conference on*, July 2011, Service migration, pp. 541–548.

[18] S. Kaisler and W. Money, "Service Migration in a Cloud Architecture," in *System Sciences (HICSS), 2011 44th Hawaii International Conference on*, Jan 2011, Service migration, pp. 1–10.

[19] C. Fehling, F. Leymann, S. Ruehl, M. Rudek, and S. Verclas, "Service Migration Patterns – Decision Support and Best Practices for the Migration of Existing Service-Based Applications to Cloud Environments," in *Service-Oriented Computing and Applications (SOCA), 2013 IEEE 6th International Conference on*, Dec 2013, Service migration, pp. 9–16.

[20] C. Haley, R. Laney, J. Moffett, and B. Nuseibeh, "Security requirements engineering: A framework for representation and analysis," *IEEE Transactions on Software Engineering*, vol. 34, no. 1, pp. 133–153, Jan 2008.

[21] D. Mellado, E. Fernández-Medina, and M. Piattini, "A common criteria based security requirements engineering process for the development of secure information systems," *Computer standards & interfaces*, vol. 29, no. 2, pp. 244–253, 2007.

[22] D. Mellado, E. Fernandez-Medina, and M. Piattini, "Applying a security requirements engineering process," in *Computer Security–ESORICS 2006*. Springer, 2006, pp. 192–206.

[23] T. M. Hesse, S. Grtner, T. Roehm, B. Paech, K. Schneider, and B. Bruegge, "Semiautomatic security requirements engineering and evolution using decision documentation, heuristics, and user monitoring," in *Evolving Security and Privacy Requirements Engineering (ESPRE), 2014 IEEE 1st Workshop on*, Aug 2014, pp. 1–6.

[24] D. S. Herrmann, *Using the Common Criteria for IT security evaluation*. CRC Press, 2002.

[25] D. C. Latham, "Department of defense trusted computer system evaluation criteria," *Department of Defense*, 1986.

[26] K. Rannenberg, "Recent development in information technology security evaluation-the need for evaluation criteria for multilateral security." in *Security and control of information technology in society*, 1993, pp. 113–128.

[27] C. Wagner, A. Hudic, S. Maksuti, M. Tauber, and F. Pallas, "Impact of critical infrastructure requirements on service migration guidelines to the cloud," in *Future Internet of Things and Cloud (FiCloud), 2015 3rd International Conference on*, Aug 2015, pp. 1–8.

[28] A. Hudic, M. Tauber, T. Lorünser, M. Krotsiani, G. Spanoudakis, A. Mauthe, and E. R. Weippl, "A multilayer and multitenant cloud assurance evaluation methodology," in *Cloud Computing Technology and Science (CloudCom), 2014 IEEE 6th International Conference on*, Dec 2014, pp. 386–393.

[29] M. Mulazzani, S. Schrittwieser, M. Leithner, M. Huber, and E. R. Weippl, "Dark clouds on the horizon: Using cloud storage as attack vector and online slack space," in *20th USENIX Security Symposium, San Francisco, CA, USA, August 8-12, 2011, Proceedings*, 2011.

[30] H. Takabi, J. B. D. Joshi, and G. Ahn, "Security and privacy challenges in cloud computing environments," *IEEE Security & Privacy*, vol. 8, no. 6, pp. 24–31, 2010. [Online]. Available: http://dx.doi.org/10.1109/MSP.2010.186

[31] I. O. for Standardization and I. E. Commission, "Information technology – Security techniques – Information security management systems – Requirements)," International Organization for Standardization, Standard, July 2005.

[32] Nist and E. Aroms, *NIST Special Publication 800-53 Revision 3 Recommended Security Controls for Federal Information Systems and Organizations*. Paramount, CA: CreateSpace, 2012.

[33] M. Flittner, S. Balaban, and R. Bless, "CloudInspector: A Transparency-as-a-Service Solution for Legal Issues in Cloud Computing," *IEEE International Conference on Cloud Engineering Workshop*, 2016.

[34] "Payment Card Industry (PCI) Data Security Standard," Nov. 2013, https://www.pcisecuritystandards.org.

[35] I. Sedinic and Z. Lovric, "Influence of established information security governance and infrastructure on future security certifications," in *Information Communication Technology Electronics Microelectronics (MIPRO), 2013 36th International Convention on*, May 2013, pp. 1111–1115.

Security Assessment Methodology for Multi-layered Clouds

The paper can be found online under the following link: `http://www.sciencedirect.com/science/article/pii/S0167404817300627`

The paper "Security Assessment Methodology for Multi-layered" was published in the Elsevier Computer and Security Journal.

Security Assurance Assessment Methodology for Hybrid Clouds

Aleksandar Hudic,[a,1,*], Paul Smith,[a,1], Edgar R. Weippl[b,2]

[a] AIT Austrian Institute of Technology,Donau-City-Straße 1, 1220, Vienna, Austria
[b] SBA Research,Favoritenstraße 16, 1040, Vienna, Austria

Abstract

The emergence of the cloud computing paradigm has altered the delivery models for ICT services. Unfortunately, the widespread use of the cloud has a cost, in terms of reduced transparency and control over a user's information and services. In addition, there are a number of well-understood security and privacy challenges that are specific to this environment. These drawbacks are particularly problematic to operators of critical information infrastructures that want to leverage the benefits of cloud. To improve transparency and provide assurances that measures are in place to ensure security, novel approaches to security evaluation are needed. To evaluate the security of services that are deployed in the cloud requires an evaluation of complex multi-layered systems and services, including their interdependencies. This is a challenging task that involves significant effort, in terms of both computational and human resources. With these challenges in mind, we propose a novel security assessment methodology for analysing the security of critical services that are deployed in cloud environments. Our methodology offers flexibility, in that tailored policy-driven security assessments can be defined, based on a user's requirements, relevant standards, policies, and guidelines. We have implemented and evaluated a system that supports online assessments using our methodology, which acquires and processes large volumes of security-related data without affecting the performance of the services in a cloud environment.

Keywords: assurance, cloud computing, security assessment, security metric, openstack,

1. Introduction

Cloud computing is a utility computing paradigm that has transformed the ICT services deployment landscape. It is used to offer scalable, highly available, on-demand, and ubiquitous services, which extend beyond single administrative and geographical boundaries. Numerous benefits, such as the ability to rapidly deploy a service, pay for resources per use, and scale them on demand, have attracted many to deploy their services in the cloud. Nevertheless, there are critical services (e.g., in the financial, electricity, water, transportation, and health sectors), which have stringent security and resilience requirements, that have not seen wide-scale deployment in the cloud. The disruption of these services could result in significant societal and economic losses [1].

To host critical infrastructure services in the cloud, a cloud provider needs to offer guarantees, which can be assured, in terms of security and resilience. Unfortunately, existing best practices for security certification and auditing provide only a snapshot of conditions. However, in elastic cloud environments, services are often subjected to continuous refinements and unpredictable changes, which can change the security and resilience posture of a service, and invalidate certificates. In this setting, re-certification may be required that has high costs and overheads. Consequently, critical infrastructure service providers are reluctant to migrate their services to the cloud [2, 3, 4]. Hence, security has become one of the most widely-considered areas of study when it comes to the Cloud computing [5, 6].

To address these shortcoming, a number of approaches have been proposed for systematic and comprehensive cloud security certification [7, 8, 9, 10, 11, 12, 13, 14]. These solutions support continuous security monitoring of the cloud. However, they can expose a cloud service providers intellectual property as part of the assessment process. A way to address this problem is to aggregate assessment results into *assurance levels*. This approach is used in Common Criteria [15]. Common Criteria assurance levels indicate the level of sophistication of the tests that have been performed on a system – functional, structural, methodical, semi-formal and formal. A further challenge is performing security assessments for services that are deployed across multiple administrative and geographical domains. For example, Dropbox [16] is a distributed storage service that uses Amazon's S3 storage for storing user's data, and therefore has independent layers that are owned by different stakeholders. Seamless and transparent service aggregation across a federation of public or private cloud service providers is therefore a vital strategy for future service provisioning that is commonly referred to as a hybrid cloud model[17]. In such a multi-stakeholder scenario, a detailed security assessment requires the support

*Corresponding author at AIT Austrian Institute of Technology
[1] firstname.lastname@ait.ac.at
[2] firstname.lastname@sba-research.org

of each stakeholder, which can be challenging to acquire, for a number of reasons.

These shortcomings indicate the need for new approaches to security certification for cloud environments that provide a current (or continuous) security assessment and, at the same time, protect cloud providers' intellectual property. In this article, we present a solution, including novel concepts, an architecture and a prototype implementation, which addresses this need. The solution offers the ability for both end customers and providers to flexibly define evaluation policies and assessment intervals. To support the assessment of services that are deployed across multiple clouds, wherein intellectual property issues arise, the details of an infrastructure are abstracted into a dependency tree model. A cloud provider's sensitive infrastructure information is protected by providing an overall assurance level to the end customer. The assessment architecture and prototype implementation are designed to flexibly scale, in order to handle the volatile workloads that are characteristic of cloud environments.

This article is structured as follows. Section 2 outlines the most relevant state-of-the-art for security assessment, certification schemes, security control approaches, and security matrices; and monitoring solutions for distributed environments. Subsequently, in Section 3, we detail the methodology of our assessment process and demonstrate it via an analytical model. Section 4 introduces the conceptual solution of the security assurance assessment model for open source cloud environments. Moreover, in Section 5, we present the implementation details and solution of our assessment model for OpenStack Cloud. In Section 6 we evaluate and argue the benefits and drawbacks of the proposed solution. Finally, in Section 7 we conclude this work.

2. Related Work

In this section, research on security assessment concepts that are related to our work are discussed. In particular, we focus on security certification schemes, security metrics and monitoring solutions for the cloud.

2.1. Security Certification Schemes

One of the most prominent approaches to security certification is Common Criteria [15], which is aligned with the ISO 15408 [18] and ITSEC [19] standards. The Common Criteria assessment approach offers a level of confidence that a set of security requirements are met by a product, which is formally referred to as the Target of Evaluation (ToE). Two further standards extend ISO 15408: ISO/IEC TR 19791 [20] focuses on operational systems, whereas ISO 18045 [21] defines the minimum actions to be performed by an evaluator, in order to conduct an ISO/IEC 15408 evaluation.

Security compliance testing for multi-layered cloud environments, which includes both auditing and certification, has been considered in recent years. The European CUMULUS[3] and CIRRUS[4] research projects aimed to address this issue. The CUMULUS project defined an extensive certification model for cloud environments [7, 22]. Some of the early certification approaches [8] aimed at providing a high level of assurance. The authors demonstrate their approach using a service-oriented architecture, which detects changes of a service with regard to security, and proposed a certification model for autonomic cloud computing systems that is based on block-based security certification [9]. In more recent work [11], the authors demonstrate the application of their certification models on OpenStack[5]. Krotsiani *et al.* [12, 13], proposed a systematic certification methodology, which is capable of certifying different types of cloud services, including IaaS, PaaS and SaaS services. In their approach [12], they detail how evidence that is used for assessment and verification of security are acquired through monitoring. Furthermore, the authors demonstrated the efficiency of their certification methodology in follow-up work [13], where they present the certification of non-repudiation in cloud storage services. Furthermore, Katopodis *et al.* [14] argue for a hybrid certification model, which is capable of supporting automation of certification processes in cloud environments, by combining testing methodologies and monitoring concepts to enforce trustworthiness and security.

Such *et al.* [23] performed an extensive assessment of assurance techniques by highlighting the required time, personnel-count, expertise, effectiveness and costs to perform various assurance techniques. The survey results indicate that the majority of investigated assurance techniques can be completed within 10 days, with a range between 2-10 days.

By parallelizing the security verification process, Khan and Hamlen [24] aim to optimize the processing costs of their model. In [25], the authors develop a framework that performs certification of a cloud topology, without revealing the underlying infrastructure – the approach is based on cryptographic primitives that verify security properties. Furthermore, Bleikertz and Gross provide an automated analysis of volatile virtualized infrastructures to check misconfiguration [26]. The authors of [27, 28, 29, 30] highlight the importance of dynamic certification for multi-layered cloud environments to verify security concerns across an infrastructure, by offering concepts based upon monitoring, security matrices and continuous assessment.

Whilst these solutions represent important work, they limit the flexibility that users have to define their security assessment policies, or are limited to preselected standards and best practice requirements. Importantly, many of the aforementioned approaches require human intervention when performing security assessments.

[3] CUMULUS (Certification infrastructure for multi-layer cloud services) `http://www.cumulus-project.eu/`

[4] Certification, InteRnationalisation and standaRdization in cloUd Security `http://www.cirrus-project.eu/`

[5] OpenStack – `https://www.openstack.org/`

2.2. Security Metrics

An essential building block for assessing any ICT asset are suitable metrics, which define the means and scope of an assessment. Savola *et al.* [31] have performed an extensive literature survey to create a security metric taxonomy. Based upon this analysis, the authors constructed a security metric for distributing messaging systems [32], supported with a risk-driven assessment [33]. Luna *et al.* [34] proposed one of the first conceptual solutions for security metrics in the cloud. However, the solution only offers a high-level systematic overview. Heyman *et al.* [35] propose security-based patterns and demonstrate how to assess the system as a whole service. This is done with an aggregation algorithm that iterates through various levels. Vaarandi *et al.* [36] built their security matrices using log files and addressed the big data analytics problem this creates. Caron *et al.* [37] propose a security metric using a system of bit vectors that show if some security requirement is needed at a particular node. To consider hierarchical dependencies, Kotenko *et al.* [38] built an Ontology-based metric. The automatic security analysis of Sun *et al.* [39] graphically visualizes security metrics to reduce the complexity of assessment and offer a better overview of the system.

2.3. Cloud Environment Monitoring

Monitoring is a major challenge for the cloud. An evaluation of both open source and commercial monitoring solutions for cloud by Aceto *et al.* [40] and Fatema *et al.* [41] indicates there are shortcomings with regard to scalability, interoperability, multi-tenancy, verifiable measuring, and service dependency.

The majority of monitoring solutions that have been proposed [42, 43, 44, 45, 46, 47, 48, 49] make use of agents, and focus on acquiring information from individual points of interest and storing it on centralized storage locations. The authors of [47, 50, 48] propose a hierarchical solution for segregating workload. Naik *et al.* [45] propose a framework for hybrid cloud integration, by supporting automation and integration of services across multiple distinct clouds. Furthermore, Rak *et al.* [49] introduce a solution for monitoring interoperable cloud applications. Massonet *et al.* [51] present a federated cloud monitoring solution, which allows cloud stakeholders to track any operation over their content across the cloud federation. Gonzalez *et al.* [52] focus on addressing security monitoring in multi-layered environments, by detecting deviations of defined behavior in their policies. Casola *et al.* [46] define specific metrics and associate them with security Service Level Objectives (SLOs), which are formally specified in Service Level Agreements (SLAs).

3. Security Assurance Assessment Methodology

This section introduces our security assessment methodology for performing a security analysis of systems and services in multi-layered and multi-tenant environments, as can be found in the cloud. The initial presentation of our methodology was introduced by Hudic *et al.* [53].

3.1. Decomposition and Abstraction of a Service or System

Despite the technological advancements of the cloud, they can also be characterized as being a complex set of systems and services. This complexity imposes the need to derive comprehensive and lightweight solutions for security analysis. To overcome this challenge, we propose a solution that first abstracts individual entities of an evaluated system (or a service), along with their dependencies. To illustrate the functionality of a complete multi-layered cloud environment, we refer to the cloud architectural framework that has been proposed by Schöller *et al.* [54]. The framework, besides highlighting the complexity of cloud environments, defines a structured and layered model of a cloud environment.

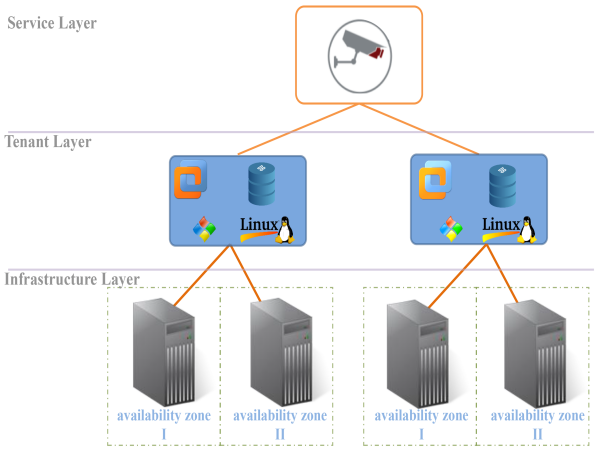

Figure 1: Simplified version of the video surveillance service that balances incoming workload, deployed in a multi-layered (i.e., infrastructure, tenant and service layer) environment and spanned across two availability zones.

For illustrative purposes, we make use of a metropolitan area video surveillance use case (i.e., traffic surveillance that includes congestion predictions, crisis traffic management, etc.) for the purpose of this work. Commonly there are cameras spread across a wide area for which they collect and transmit video footage to a central processing location. Depending on the size of a particular city the scale of available video material can vary, but it is commonly very large. Therefore we re quire an environment with large processing and storing capabilities such as cloud for hosting such services. In Figure 1 we show a web based video surveillance service that balance the processing load, in our particular case due to the simplicity we will use only two underlaying virtual machines for balancing the processing load. The load balancing is performed across two distinct availability zones set as a fail-over strategy so that in case of disruption we can assure high availability. Furthermore, the virtual machines contain software

that processes the incoming video surveillance and replicates the content on block storage drives in two availability zones.

First initial step, prior to the assessment, is to perform the holistic service abstraction by decomposing the service on its autonomous entities (i.e., video surveillance service, virtual machines, and physical storages). These entities are then illustrated via hierarchical interdependencies which depict a general tree model [55], Figure 2. Essentially, as shown in Figure 2, we illustrate a video surveillance service which resides on top of two virtual machines, that are referencing object storage across two availability zones (physical storage servers).

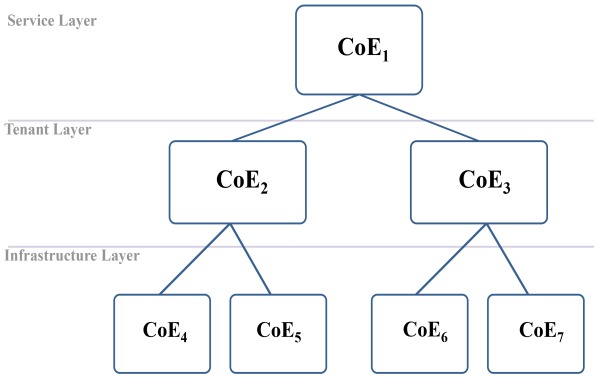

Figure 2: Holistic Abstraction of the service introduced in Figure 1 on its autonomous components. We formally refer to the whole abstracted service as Target of Evaluation (ToE), and to its individual components as Components of Evaluation (CoE).

Since we aligned our holistic service abstraction model with the notation of the Common Criteria [15], to support the our security assurance assessment approach, below we outline more detailed common criteria based components of our approach:

Target of Evaluation (ToE) - is a system or a service (in our particular case the video surveillance service) illustrated as a hierarchical and abstract composition of individual interdependent entities, which we refer to as components of evaluation (CoE), Figure 2. An individual ToE can be composed from one or more components, Equation 1, or one or more group of components (GoE), Equation 2.

$$ToE_A = \{CoE_1, CoE_2, ...CoE_N\} \tag{1}$$

$$ToE_A = \{GoE_a, GoE_b, ...GoE_z\} \tag{2}$$

Component of Evaluation (CoE) - is a single autonomous entity of a system or a service (e.g., physical servers, virtual machines, service hosted on top, etc.) that is being evaluated, Figure 2. An individual component CoE_x can be an element of more than one group of components (i.e., GoE_A, GoE_B, *or* GoE_Z), Equation 3, and even multiple services (i.e., ToE_1, ToE_2, or ToE_N), Equation 4.

$$CoE \subset GoE_i, \in \{1, 2, ...N\} \tag{3}$$

$$CoE \subset ToE_i, \in \{1, 2, ...N\} \tag{4}$$

Group of Evaluation of Evaluation (GoE) - is a composition of components or a subset of entities of a service (e.g., group of physical servers or virtual machines that operate under same security conditions), Equation 6, that is of a particular interest to a user, Figure 4. A GoE can be a subset of one or more ToE, Equation 5 (e.g., we can consider as our GoE all physical servers running on the infrastructure level that can host multiple ToE). Furthermore, a CoE can be a part of multiple GoE simultaneously, i.e., database that stores information from several different services, consequently there may be overlappings in terms of multiple CoE between different GoE, Equation 7.

$$GoE \subset \{ToE_1, ToE_2, ...ToE_N\} \tag{5}$$

$$GoE_a = \{CoE_1, CoE_2, ...CoE_N\} \tag{6}$$

$$GoE_a \cap GoE_b \cap GoE_c \cap ... \cap GoE_z \neq \emptyset \tag{7}$$

Security Property (SP) - is a parametrized security requirement used to validate qualitatively the security of a particular ToE, i.e., blueprint for measuring a security requirement like encryption in our service across multiple abstraction layers. Each security property is defined to validate security across each individual element of a service consistently. Major challenge that we confront when designing and individual SP is the ability to uniformly measure the SP across various component types and levels. Example of a SPs that we defined and developed under the scope of SECCRIT project is illustrated in Table 1.

Security Property Vector (SPV) - is a composition of SP in form of a bitwise vector, Equation 9. The bitwise representation of properties in the SPV is used to support the bitwise conjunction within our security assurance assessment process to validate the presence of SPV's properties across the whole service. Also for the SPV, it is mandatory that it is used consistently without changing the order and number of security properties during the evaluation, Equation 10. One of the essential SP for our use case, SP_2 - Information (Data) Consistency, is used to validate that there was not alteration of the original video surveillance material.

Assurance Security Class (SC) - is a cluster of various SPs that undertake the common scope or purpose, in our case security class (e.g. confidentiality, integrity or availability), Table 1.

Evaluation Set(ES) - is an encountered set of distinct security properties, Equation 8, used for validating security conditions of a service(i.e. set of properties that we want to validate across the infrastructure to protect our video surveillance data). The ordering of the SPs, regardless of the security class clustering is the foundation for

Table 1: Assurance security properties and corresponding classes analyzed and developed under the scope of EU FP7 research project SECCRIT.

Assurance Class	Security Property
Confidentiality	SP_1 - Concurrent session control
	SP_2 - Password Rotation
	SP_3 - Strong Password
	SP_4 - Encryption
Integrity	SP_1 - System/Service Integrity
	SP_2 - Information (Data) Consistency
	SP_3 - Alteration Detection
	SP_4 - Error Correction
Availability	SP_1 - Geo-location
	SP_2 - Service Availability
	SP_3 - Service Isolation

supporting different policies that we address later, e.g. Table 3 and Table 4.

$$ES = \{SP_1, SP_2, ...SP_K\} \tag{8}$$

$$SPV = [\{0/1\}_1, \{0/1\}_2, ...\{0/1\}_K] \tag{9}$$

$$\forall CoE \in ToE \models |SPV| = |ES| = K \tag{10}$$

Assurance Level (AL) - is a numeric value that represents qualitative security assurance level of a particular entity assessed with respect to predefined security conditions, i.e., security properties. Security conditions are in our case defined in form of an evaluation set for particular component, group, level or target of evaluation. The assurance level model is adhered by the Common Criteria approach, and extended to support holistic security assurance model, Equation 11.

$$AL = j, j \in 1, 2, 3, 4, 5, 6, 7 \tag{11}$$

3.2. Assessment, Aggregation and Computation of Security Assurance Level

The concepts for holistic service abstraction and decomposition defined in Section 3.1 are used as the foundation of this section to build security aggregation policies, defining security properties and finally the security assurance aggregation algorithm.

3.2.1. Aggregation Policies

We define aggregation policies that distinguish various security conditions through the security aggregation process and making decisions that support the tree traversal algorithm, explained afterwards, for delivering the overall security assurance result.

Uniform Aggregation Policy - defines rules to perform a uniform bitwise conjunction of a predefined and unique SPV across each individual CoE in a ToE, Equation 12 and Equation 10. The aggregation can be successfully performed only if during the process of iterating through each element of the ToE, SPV is being consistently used as defined by Equation 10. As the final step of the assessment process the aggregated or resulting SPV is associated with the corresponding assurance level defined by assurance level policy (the assurance level associations, e.g. Table 3 and Table 4).

$$\begin{gathered}\mathcal{A}(ToE) \vdash \forall CoE \in ToE \Rightarrow \\ SPV(CoE_1) \wedge SPV(CoE_2) \wedge ... \wedge SPV(CoE_N)\end{gathered} \tag{12}$$

Level Aggregation Policy - defines rules to perform the process of aggregating security conditions per individual components across an abstraction level, Equation 16 to deliver the security assurance level, Equation 17. Furthermore, the components within an abstraction level do not necessary need to reside under the jurisdiction of a single parent. In that particular case, the assurance aggregation process is conducted by iterating through each individual element of the abstraction level, Equation 10. An abstraction level is a composition of components that are not necessarily a subset of an only one evaluation service, Equation 14. Within the set of components that corresponds to an individual abstraction level, there cannot be a single component that is a member of another subset or abstraction level, Equation 13. This policy prevents that we aggregate across multiple levels, i.e., for example if in case of aggregating security conditions across a physical level where only servers are taken in to consideration a virtual machine is included by accident. A ToE can be composed of multiple abstraction levels, Equation 18, ordered hierarchically where we determine the overall security assurance level by taking into consideration each individual level in the hierarchy.

$$L_1 \cap L_2 \cap ... \cap L_K = \emptyset \tag{13}$$

$$L_K \subset ToE_i, \in 1, 2, ...N \tag{14}$$

$$\mathcal{A}(ToE) = \{\mathcal{A}(L_1), \mathcal{A}(L_2), ...\mathcal{A}(L_K)\} \tag{15}$$

$$L_K = \{CoE_1, CoE_2, ...CoE_X\} \tag{16}$$

$$\begin{gathered}\mathcal{A}(L_K) \vdash \forall CoE \in L_K \Rightarrow \\ SPV(CoE_1) \wedge SPV(CoE_2) \wedge ... \wedge SPV(CoE_X)\end{gathered} \tag{17}$$

$$ToE = L_1 \cup L_2 \cup ... \cup L_K \tag{18}$$

Group Aggregation Policy - defines rules to perform the process of aggregating for a predefined set of security conditions across a subset of components that are of a particular interest, Equation 21.

It is mandatory that a SPV is being consistently used across each individual component within an interest group, Equation 10, whereby an additional condition is also valid

ToE = GoE. Unlike the level aggregation policy an individual component of a GoE can be present in more than one GoE, Equation 20. Within the scope of a single evaluation service we can define multiple groups of interest, Equation 22, whereby each group is defined with a subset of components that are under the juristiction of a single evaluation service component set, Equation 23.

$$GoE_K = \{CoE_1, CoE_2, ...CoE_X\} \tag{19}$$

$$GoE_1 \cap GoE_2 \cap ... \cap GoE_K \neq \emptyset \tag{20}$$

$$\begin{gathered}\mathcal{A}(GoE_K) \vdash \forall CoE \in GoE_K \Rightarrow \\ SPV(CoE_1) \wedge SPV(CoE_2) \wedge ... \wedge SPV(CoE_X)\end{gathered} \tag{21}$$

$$ToE = GoE_1 \cup GoE_2 \cup ... \cup GoE_K \tag{22}$$

$$GoE_X \neq ToE_1 \cap ToE_2 \cap ... \cap ToE_K \tag{23}$$

3.2.2. Security Property Policies

To ensure the consistency of an aggregation process across each evaluated component within a service concise definition of each property in a SPV is mandated. Thus, a reliable bitwise conjunction of a predefined security property evaluation set across the target of evaluation can be performed. Each evaluated SP is defined by a concise policy that distinguishes SP logical states (e.g. 0 in case when a particular security property condition are not fulfilled or 1 when being fulfilled), which we refer to as *Security Property Policy* (SPP).

Table 2: Security property policy for security property encryption

Logical State	**Infrastructure level**	**Tenant level**	**Service level**
0	Some disks or partitions are not being encrypted	Some disks or partitions are not being encrypted	There is at least one port open, except 443, and used as communication channel
1	All disks and and partitions encrypted ⇒ Disk encryption 100%	All disks and and partitions encrypted ⇒ Disk encryption 100%	HTTP protocol over TLS/SSL, all communication ports closed except 443 used to direct all communication

Due to the space limitation we only illustrate one representative example of our SP, the security property *Encryption* ($SP_{encryption}$), shown in Table 1. We define encryption as a security property that supports confidentiality by mandating minimum configuration of a cryptographic mechanism applied on a particular evaluated entity. In Table 2 we detail per each individual level and logical state conditions of $SP_{encryption}$. Each individual SP in security assurance framework, like $SP_{encryption}$, is defined in line with state of the art standards and best practices:

- NIST SP 800-111 - Guide to Storage Encryption Technologies for End User Devices [56]
- NIST SP 800-127 - Guide to Securing WiMAX Wireless Communications[57]
- NIST SP 800-12 - An Introduction to Computer Security: The NIST Handbook [58]

In addition, for each SP we outline which monitoring artifacts across layers is being verified to support a particular SP like encryption:

- Infrastructure layer: physical disks are validate to check if the disk has been completely encrypted according to standards and best practices [56, 58], i.e., validate encryption properties (e.g., key length > 128 bit and encryption type == AES) by validating information of all available or specified block devices connected to a particular component (virtual or physical server).
- Tenant layer: virtual disks are validate to check if the disk has been completely encrypted according to standards and best practices [56, 58], i.e., validate encryption properties (e.g., key length > 128 bit and encryption type == AES) by validating information of all available or specified block devices connected to a particular component (virtual or physical server).
- Service layer: communication channels are validated to if the proper encryption mechanisms are being used in line with standards and best practices [57, 58], i.e., validate that all port 80 is closed and all communication is diverted to SSL port 443.

3.2.3. Aggregation Algorithm

We leverage in our security aggregation process, described in Algorithm 1, Boolean algebra as the foundation of our security aggregation method. We perform bitwise conjunction of individual SP bit in SPV for each evaluated CoE in a ToE by performing a post order tree traversal procedure that iterates through a general tree. The tree traversal aggregation algorithm is initiated by requiring as input the following three parameters: root CoE of an evaluated service, security property evaluation set and aggregation policy. During the aggregation process it is

important that for each component of an evaluated service SPV is consistently used, Equation 10. Furthermore, the tree traversal is carried out via the recursive procedure that performs post order general tree traversal that as returns as the result the overall assurance $\mathcal{A}(ToE)$. In addition, our aggregation algorithm differentiates two main aggregation aspects:

- *Horizontal Aggregation* - bitwise conjunction of components that reside under the jurisdiction of a singe parent (i.e. virtual machines since they are both at the same logical level, Figure 2, the aggregation of VM_1 (CoE_2) and VM_2 (CoE_3)).
- *Vertical Aggregation* - aggregation which commonly occurs in a form of child-parent relationships or the bitwise conjunction of the result of horizontal aggregation and corresponding parent (i.e. conjunction of the result of the aggregation of VM_1 (CoE_2) and VM_2 (CoE_3) and the root service component (CoE_1), Figure 2).

Algorithm 1 Tree Traversal Assurance Aggregation

Require: CoE_{root} , ES, $\mathcal{P}$
Ensure: $\forall CoE_i \in ToE \models |ES| = |SPV|$, $i \in 1, 2, ...N$
  traverse (CoE_{parent}, $\mathcal{P}$)
    if $CoE_{parent} \models \mathcal{P}$ **then**
      initialize SPV_h, SPV_v
      for each CoE_{child} in CoE_{parent}, from left to right **do**
        if CoE_{child} has children **then**
          traverse all children
          SPV_v = **traverse**(CoE_{child}, $\mathcal{P}$)
          vertical aggregation
          $\forall SPV_v \wedge SPV_{CoEchild}$[i], $i \in 1, ...N$
          horizontal aggregation
          $\forall SPV_h[i] \wedge SPV_v$[i], $i \in 1, ...N$
        else
          horizontal aggregation of the leaf nodes
          $\forall SPV_h[i] \wedge SPV_{CoEchild}$[i], $i \in 1, ...N$
        end if
      end for
      return SPV_h
    end if

3.2.4. Assurance Level Association Policy

In order to define a qualitative value of security, as a security metric, we we define supporting policies that associate the results of aggregation to a corresponding assurance level. Thus, not only the resulting SPV from the assurance aggregation can be associated with the level, but also for each group, level or component we can provide associating assurance level. However, to associate the assurance level to a resulting SPV we have to first order or prioritize and define SPP for each SP. The priority of SPs can be either defined by a customer or aligned with standards and best practices. The left hand side of the Table 1 illustrates an example of security property prioritization, whereby the properties have been additionally clustered and prioritized per individual class. Respectively, each of the security classes (confidentiality, availability and integrity) can be additionally prioritized to support our assurance policies. Finally, after the evaluation set is prioritized, logical state policies of each SP are set (i.e. defining security property policy), a final security metric for associating the result security property vector with an assurance level can be aligned, the *Assurance Level Association* shown in the right hand side of the Table 1. We adhere also the the Common Criteria approach that associates 7 distinct assurance levels in our model, as shown in Table 1.

Table 3: Assurance level association policy shows how individual SPV are associated with the corresponding assurance level.

Security Property Vector (SPV)				**Assurance level association**
SP_N	SP_3	SP_2	SP_1	AL
0	0	0	0	-
0	0	0	1	AL1
0	0	1	0	AL2
0	0	1	1	AL2
0	1	0	0	AL3
0	1	0	1	AL3
0	1	1	0	AL4
0	1	1	1	AL4
1	0	0	0	AL5
1	0	0	1	AL5
1	0	1	0	AL6
1	0	1	1	AL6
1	1	0	0	AL7
1	1	0	1	AL7
1	1	1	0	AL7
1	1	1	1	AL7

On the left hand side of the Table 3 there are all potential combinations for a particular set of 7 SP (i.e. $2^{|SP|}$ potential combination) that are been selected only for illustrative purposes from Table 1. Due to the fact the number of potential combinations is exponentially rising with the increase of security properties, we illustrate the individual assurance levels by using ranges to reduce the scale of a table, as demonstrated by the assurance association for our evaluation model in Table 4.

3.2.5. Protection Profile

Unlike in case of Common Criteria, we enhance the protection profile to also support the perspective of a whole service (e.g., video surveillance service), domain (e.g., health

care data protection) or a group and not only on the component basis. A protection profile in our framework is defined by the security property policy, evaluation set and it can be facilitated based upon a service, component, group, type of component, or with regards to a particular domain.

3.3. Evaluation of the Assurance Aggregation Model

We will again refer to the video surveillance service use case introduced in the beginning of this section, Figure 1, to demonstrate our security assurance assessment methodology.

First, as shown in Section 3.1, we perform abstraction of our video surveillance service, Figure 1, to build a hierarchical abstraction tree model, Figure 2. To each individual CoE_K of our tree model, we assign a $SPV(CoE_K)$ as the representation of the current security status for a particular CoE, as shown by Figure 3. Then, we define our evaluation set as follows, i.e. set of properties that we are going to validate across our tree model(evaluated service): ES = { SP_7 = Encryption, SP_6 = SystemService Integrity, SP_5 = Concurrent Session Control, SP_4 = Password Rotation, SP_3 = Strong Password, SP_2 = Alteration Detection, SP_1 = Error Correction}. For the purpose of the evaluation model we associated with the CoE_1, CoE_2 and CoE_3 SPV that show deficiencies for alteration detection, strong password, password rotation and concurrent session control.

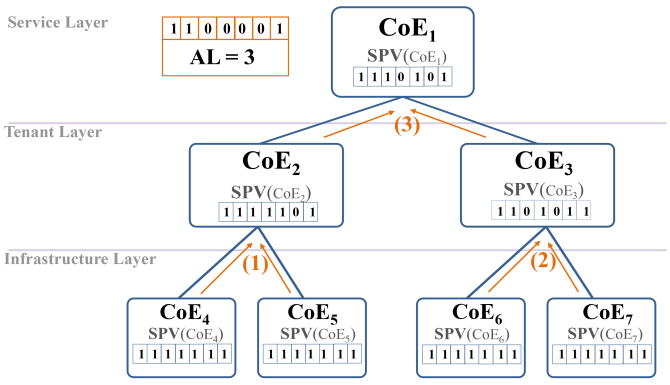

Figure 3: Illustration of security aggregation process across the abstracted service with individual steps in line with our aggregation algorithm

The abstracted service tree is the blueprint of our postorder tree traversal iteration algorithm (Algorithm 1). In Figure 3, we illustrate main aggregation steps of our algorithm performed during the aggregation process for this particular use case. *Step 1*, our Algorithm 1 initiates the aggregation process by performing an horizontal aggregation of the SPV(CoE_4) and SPV(CoE_5), starting from CoE_4 as the most left child of our ToE. Afterwards, the result of the horizontal aggregation is vertically aggregated with the parent component of CoE_4 and CoE_5, CoE_2. *Step 2*, horizontally aggregates the SPV(CoE_6) and SPV(CoE_7), starting from CoE_6 as the most left child of the parent component CoE_3. Then it vertically aggregates the result of the horizontal aggregation with their parent component CoE_3. Finally, *Step 3* first performs the horizontal aggregation of CoE_2 and CoE_3 and then the result of the horizontal aggregation is vertically aggregated with the root component CoE_1. The final result gives us an AL(ToE_2) = 3 according to the Assurance level association Table 4. Since in our tree traversal algorithm we always include an aggregation policy, the default aggregation policy, if not explicitly stated, is the uniform aggregation policy that includes all components of the service.

Table 4: Assurance level association in case of large amount of SP, e.g. 7SP, translated into hexadecimal representation.

	SPV ranges	
	Decimal	Hexadecimal
AL1	1 - 15	1 - F
AL2	16 - 31	10 - 1F
AL3	32 - 63	20 - 3F
AL4	64 - 95	40 - 5F
AL5	96 - 111	60 - 6F
AL6	112 - 123	70 - 7B
AL7	124 - 127	7C - 7F

Moreover, if using GoE we are able to isolate a custom subset of components for independent and simultaneous assessment. The application of such isolation is derived through specific policies (e.g. Uniform Aggregation Policy, Level Aggregation Policy or Group Aggregation Policy) that we include in our algorithm. We demonstrate in our use case the usage of the group aggregation policy, shown in Figure 4, by isolating two subsets of components, $GoE_1 = \{CoE_1, CoE_2, CoE_4, CoE_5\}$ and $GoE_2 = \{CoE_1, CoE_3, CoE_6, CoE_7\}$. The tree traversal assurance aggregation algorithm in case of group policies initiates an independent traversal process for each GoE. Hence, the traversal algorithm only performs aggregation only on a particular subset of components that correspond to a particular group. In case of GoE_1 the resulting SPV'(GoE_1) = 1110101 and in case of GoE_2 the resulting SPV'(GoE_2) = 1100001. We can see that due to the differentiation in configuration of VMs (i.e., CoE_2 and CoE_3) the end resulting SPVs differentiate themselves as well. Therefore the we have for the first group of interest, AL(GoE_1) = 6 and for the second group of interest AL(GoE_2) = 3.

4. Security Assurance Assessment Model for Open-source Cloud Environments

The security assessment commonly requires intensive acquisition of security related information during the assessment process, unfortunately this is restricted by most of the service providers to avoid any infrastructure related information exposure. Having that in mind, we design the security assurance assessment methodology framework

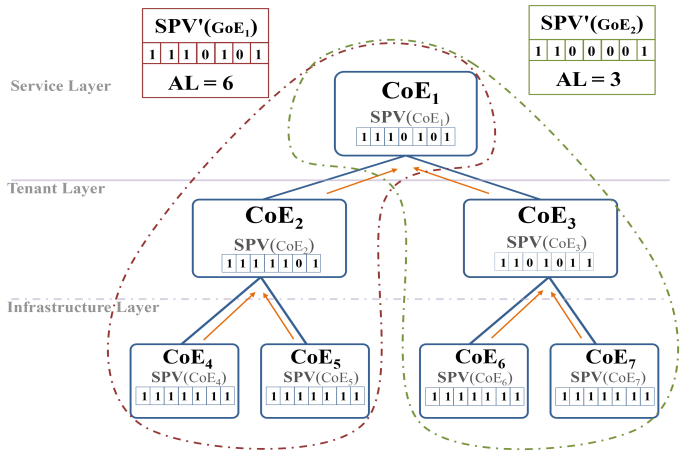

Figure 4: Identification and isolation of specific components that are of particular interest with group evaluation policy, GoE_1 and GoE_2, for which we calculate independent assurance level

that offers a transparent and nearly real-time security assessment without exposing provider's internal sensitive information and yet offering a security assessment of services according to user's requirements. Our model limits the information that the external parties have access to and offers only abstracted security related information, defined by the trust boundary marked with red dashed line in in Figure 5.

Due to the limitations and restrictions imposed by most commercial cloud providers we focus, but not limit, our solution to the open source cloud technology that gives us the liberty to obtain detailed security related information across the whole cloud infrastructure (i.e., each abstraction level - infrastructure, tenant or service).

Our solution is designed as coherent system composed of autonomous modules that can perform lightweight security assessment independent of the cloud platform, and can be deployed inside of the cloud platform that we are performing assessment and externally as a third party provisioning. Therefore we distinguish the deployment boundary marked with green dashed line in in Figure 5, because right hand side with regards to deployment boundary can be deployed inside the cloud and at third party service provider. The complexity and large scale of a cloud environments imposes to process a vast amount of information in case of any kind of an assessment. Therefore, we design and implement independent functional subsystems to achieve high availability and scalability of our concept. These modules are depicted in Figure 5 as individual sub-systems that perform the following functional sub-systems: *information acquisition sub-system, messaging sub-system, processing sub-system, storage and assurance computation sub-system, and presentation sub-system.* Furthermore, we design the security assurance assessment framework to operate in the following three operational modes:

- *continuous* by performing the security assurance assessment in discrete predefined periodic time intervals, which we refer to as collecting cycles,
- *on demand* by performing the security assurance assessment at certain points based upon users needs (i.e. during audits),
- *event based* by performing the security assurance assessment at the point when a change in the system occurs (i.e. when certain collector detects that at least one SP has changed its state).

The use case proposed at Section 3 in Figure 1 is now used as the information acquisition sub-system, shown in Figure 5 on the most left side. This is the first sub-system of our assurance assessmend model Then and also the only one that requires to be directly integrated in to the cloud environment. The security related information of an individual component are being acquired via small lightweight services called *collectors*. The scope of information being collected is defined security requirements defined prior to the assessment process, which we refer to as evaluation set of security properties introduced in 3 section. Each component's collector formats the acquired information and emits it to the second sub-system of our assessment framework, the *messaging sub-system*. Due to the fact that in environments such as cloud our services a most likely to be composed of vast amount of components that would emit simultaneously collected information, and therefore our system is challenged with large amounts of data that needs to be systematically organized per service and corresponding components. Therefore, we integrate in our design a distributed messaging system capable of structuring large amount of incoming messages to avoid any potential information lesses. In addition, regardless of the incoming message workload the messaging system offers scalability, fault tolerance, and guaranty for delivering messages. Next, messages delivered to our messaging system have to be processed by the next sub-system of our assessment framework, the *processing sub-system*. Due to the large amount of messages we require a system capable of processing those messages capable of parallelizing its processes as a distributed processing system. The main role of the processing system is two fold: auditing all incoming messages in to a storage system and simultaneously processing those messages. To perform these processes simultaneously both auditing and transformation process should be instantiated with the same particular message. Reliability for processing messages is ensured by tracking a process life-cycle and re-initiating it in case of failure. The output of both auditing and transformation process is stored to our next subsequent sub-system, *storage sub-system*. After the raw information has been transformed to a bitwise vector SPV, and the SPV is stored the storage sub-system, the assurance level computation process calculates the assurance level. Finally, after the security assurance level is calculated it is being presented by the last sub-system of our solution, *presentation sub-system*.

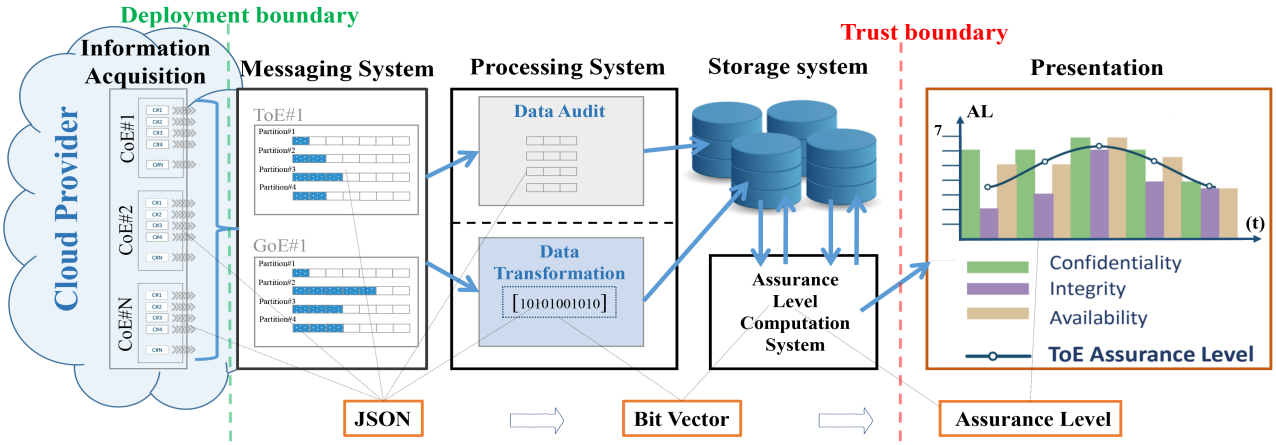

Figure 5: Security assurance assessment framework architecture with independent sub-systems (information acquisition sub-system, messaging sub-system, processing sub-system, storage and assurance computation sub-system, and presentation sub-system) used to perform message collection, processing, assurance level calculation and presentation.

5. Prototype Implementation

This section highlights the motivation for implementing our security assurance assessment framework.

5.1. Motivation

To address the problem of acquiring, transmission and processing large scale security related information across a cloud environment a solution capable of handling such high intensive workloads is required. Having these objectives in mind, we built a solution that is technology independent, can be hosted either as third party or in internal solution, and is capable of scaling its processing capabilities.

We start with evaluating the size and structure of the messages that are going to be acquired and transmitted, which must remain as small as possible and provide sufficient level of details. Therefore, we performed an empirical analysis of different message types by taking into consideration Comma Separated Value (CSV), JavaScript Object Notation (JSON) and Extensible Markup Language (XML) formats. We generated structured messages for each of the aforementioned formats by increasing the number of security properties per message for each format type from 1 to 1000 SP per message. The outcome of our analysis, Figure 6, indicates the following average sizes of a message per SP: CSV 50 bytes, JSON 64 bytes, and XML 99 bytes.

The prior concern of our design to have minimal impact on the performance is fulfilled by the by the CSV as the best solution in terms of message size. However, since efficient processing of stream based data structures and its serialization as objects is required, JSON and XML are imposed as more eligible solutions. Further, analysis indicated that the JSON message outperforms the XML messages in terms of size and practicality for object serialization, Figure 6. Therefore, we rely on JSON message format that we use as core technology in our implementation for information acquisition scripts refer to as *collectors*.

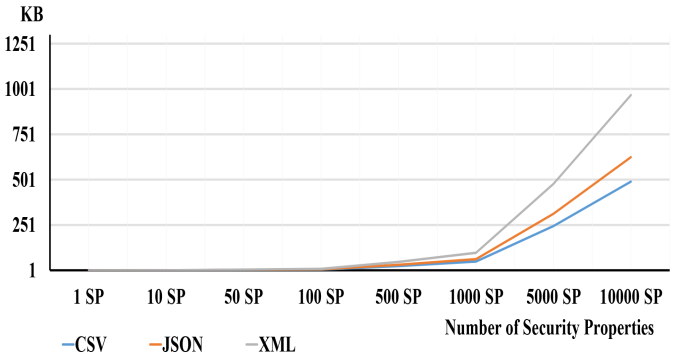

Figure 6: Evaluation of size for different message types (JSON, CSV, and XML) with respect to different count of security properties.

Collectors are pre-installed scripts at the service provider that perform data extraction, formating and transmission supported with configuration file that is used for remote configuration. We can configure the collector by controlling messaging system host (i.e. destination of messaging system clusters), operational mode of the collector (i.e. continuous, on demand, or event driven), identifiers used for later distribution and message processing (i.e. topic and component identifier). In addition we estimated the amount of the potential workload by putting into the correlation the average size of JSON message, security properties and evaluated components. The results, shown in Figure 7, indicate the exponential growth of the data that reaches easily several Gigabytes of workload that we have to take into the consideration when designing our system.

5.2. Implementation

We use the OpenStack open source cloud platform as the foundation of our research testing due to its composite and modular nature. The OpenStack cloud platform

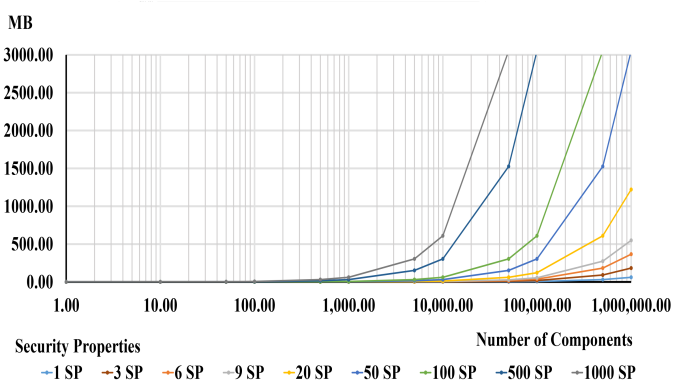

Figure 7: Estimation of the amount data being produced intended for processing in a single collecting cycle (ΔT) depending on number of evaluated security properties and number of components evaluated.

allows us to flexibility to investigate and harvest information across the complete cloud infrastructure without restrictions. In spite of the fact that the aim of this work is focused towards cloud based environments, in principle it is also applicable to a brother set of ICT environments (e.g. grids, clusters, virtual environments). The information acquisition process starts by extracting the raw data from a CoE and constructing a JSON message that is then sent to the messaging sub-system. The collector uses a supporting configuration file that mandates which properties are being monitored, where to send the information to and how often the acquisition is performed. For the messaging system we configure and implement Apache Kafka distributed messaging system [59, 60] to ensure reliable message handling. We implement the Apache Kafka services due to the ability to offer rapid scalability across multiple brokers and clusters, and high availability that is ensured through data redundancy. Apache Kafka clusters incoming messages selectively based upon the message topic identifier, and directs them to the corresponding message queues or partitions. Although that with the replication functionality we increase information availability, the replication factor unfortunately downgrades the performance and therefore in our solution we only use minimal number of copies (e.g., 2 factor replication).

Taking into consideration the extent of components in cloud environments, we require a processing sub-system capable of processing large scale data that can reliably expand and parallelize its processing capabilities and withstand the data processing rate above the input data rate. Therefore, after being commissioned in the distributed messaging sub-system, messages are provided based on publish-subscribe model the processing sub-system. The processing sub-system transforms the incoming JSON messages in bitwise vectors and simultaneously establishes an audit trail. Due to the to the high compatibility with Apache Kafka distributed messaging sub-system we integrate the Apache Storm [61, 62] as the distributed processing segment for our solution. The Apache Storm provides scalability, fault-tolerance of running tasks, and it is easily extensible to provide high performance processing. Furthermore, Apache Storm is a distributed real-time computation system for processing large volumes of high-rate data extremely fast, e.g. over a million records per second per individual node on a cluster of modest size. Raw, processsed and analyzed data is being stored into database storage system. As we can even combine the Apache Kafka as the next message storage option the storage system is not the prior concern of this work and therefore we take it under consideration for future work. After the raw data is transformed into security property bit-vectors, the assurance level computation module computes assurance level by iterating through general tree model with our algorithm 1. Finally the results are presented as continuous feedback of security status for our evaluated service to users.

Our solution is focused to preserve the privacy by protecting internal intellectual property of a cloud service provider by design. The external parties interested in validating the security conditions of a services or a system hosted in cloud, do not get direct access to sensitive information and yet they are ensured that security is being maintained according to their security requirements. Primarily, to ensure the integrity and authenticity of information being acquired, tempering of collectors and information has to be preserved during the collection and transportation. Trusted Platform Module (TPM) [63] and, its extension, Hardware Security Module (HSM) are both solutions that can provide a more secure code execution environment by validating the consistency of code being executed. The main difference lies in their appliance, TPM is an integrated micro-controller where the cryptographic keys and all computation takes place, whereby the HSM is an external module like USB which contains the cryptographic primitive used in cryptographic functions. The cryptographic functions are used to validate the consistency of executed code and information stored in a particular environments. Although, these concepts increase the code execution security they do not provide ultimate security for the user.

6. Prototype Evaluation

In this section we demonstrate the performance of our system and argue our results with supporting analysis. We support our findings with the estimation of input data workload on the messaging sub-system by taking in to the consideration number of evaluated components and SPs. In addition, we simulate the input data workload, auditing and processing capabilities on our system also with regards to number of evaluated components and SPs for three different configurations scenarios.

We performed our performance analysis Amazon infrastructure by deploying it on five EC2 instances (one instance was used for message production and measuring the performance, and four instances were used for deploying our framework; three for messaging sub-system and

one for processing sub-system) with the following configuration: c3.8xlarge instance type [6], 32 vCPU processors, 244 RAM memory, 2 x 320 GB SSD storage.

6.1. Evaluation Scenarios

We demonstrate the effectiveness of our solution with respect to a real world scenario by observing the performance for different number of components and security properties being evaluated, Table 5. We illustrate three distinct evaluation scenarios, two cloud-based and one non cloud-based scenario. The cloud based scenarios are focused on assessing components in cloud data center by using Amazon as an example, whereby we distinguish regions Frankfurt and Ireland as two independent data centers. As a third use case scenario, which is a non-cloud environment, we use our research center IT infrastructure. We define the border lines of our evaluation with respect to number of evaluated components and security properties by stating the following assumptions:

- message production during our evaluation is limited to the computational capabilities of a single amazon instance
- security assurance assessment framework is deployed on a third party infrastructure and therefore number of components that can be evaluated are limited by the publicly available IP addresses of a particular use case scenario
- number of evaluated security properties is defined based up on number of security requirements defined by security standards

According to Amazon's CTO Werner Vogels, Amazon tries to keep the size of their data centers under 100,000 servers[7]. Although it is a fairly naive approach, we take into consideration publicly available information of Amazon for IP address ranges[8] to estimate the potential amount of publicly accessible elements inside a cloud. Furthermore, we will say that an individual region is composed of a minimum one data center. Taking the public IP address ranges of an Amazon region and count of particular servers into consideration, we can easily approximate a potential number of components that could be monitored or assessed with regards to security (e.g., Amazon region Frankfurt 750,000 and Amazon region Ireland 5,000,000). In case of non-cloud use case the information the components count is determined by consolidating with our IT department. Furthermore, for defining realistic number of SP we use the following standards, guidelines and best practices as the reference: CUMULUS EU FP7 Research Project - 72 Certification Security Properties, ISO/IEC 27001 - Information security management -114 security controls [64], NIST 800-53 - 240 security requirements [65], Pay Card Industry Data Security Standards (PCI DSS) - 242 security control requirements [66] OWASP Application Security Verification Standard - 205 Requirements [67]. Our results show the correlation between the components per individual scenario vs number of security properties per components, as shown in Table 5. Furthermore, in our evaluation we highlight the key factors for performance with respect to configuration of our distributed messaging and processing systems. Therefore, we differentiate our results according to single and multi-cluster configuration setup for Apache Kafka distributed messaging system; and for Apache Storm distributed processing system audit and assurance level computation capabilities.

6.2. Results

6.2.1. Input Message Rate Simulation

We start our evaluation by evaluating the incoming workload that our system can withstand, by measuring the number of incoming messages. Thus, our tests are focusing on messaging sub-system form Figure 5. In order to evaluate how our system outperforms with regards to the workload we conducted the tests with the regards to the following three configuration setups of our distributed messaging sub-system:

1. single cluster - single broker setup, 1 Broker, 15 partitions
2. single cluster - multi broker setup, 5 Brokers, 80 partitions
3. multi cluster - multi broker setup, 15 Brokers, 300 partitions

A cluster is a physical, or in our case virtual, machine with limited amount of computational and memory resources where we run one or more Kafka brokers. We leverage the Kafka design capabilities to scale across multiple clusters and therefore extend the computational capabilities for handling incoming workload. Furthermore, a broker is a virtual abstraction that is in charge of provisioning, maintaining, and managing message queues or partitions across clusters. For each of the setups we used a separate instance for deploying message producers that simulated large scale message production. Additionally, due to our limitation of running millions of components that send the messages simultaneously we leverage batching concept to increase the number of messages that we produce. Thus, we batched messages as blocks of 4,096 and 16,384 bytes, that we will refer to as small or big batch, respectively. We have chosen these two batch size based upon the JSON message size to see how does the batch size influence the performance. In case of 4,096 batch size can observe how does the performance changes if we put multiple JSON messages in to a single batch and what happens in case when we have to

[6] Amazon instance types - `https://aws.amazon.com/ec2/instance-types/`, visited 11.03.2016

[7] Amazon Cloud infrastructure `http://datacenterfrontier.com/inside-amazon-cloud-computing-infrastructure/`

[8] Amazon Web Service - `http://docs.aws.amazon.com/general/latest/gr/aws-ip-ranges.html`

Table 5: An overview of the auditing and processing cycles time for various standards and best practices containign security control elements applied in three different application scenarios within respect of two configuration setups.

SCENARIO	CONSUMPTION	ACTION	CUMULUS 72 properties	ISO/IEC 27001 114 controls	NIST 800-53 240 requirements	PCI DSS 242 requirements	OWASP 205 requirements
			Δt				
Amazon Cloud - Region Frankfurt 750,000 Components	Single-broker - 15 Threads	Audit	4	6	13	13	10
		Comp.	6	8	19	19	15
	Multi-broker - 30 Threads	Audit	3	4	8	8	6
		Comp.	4	4	12	12	9
	FRAMEWORK ΔT		7	11	19	19	17
Amazon Cloud - Region Ireland 5,000,000 Components	Single-broker - 15 Threads	Audit	28	33	78	78	78
		Comp.	31	39	124	124	90
	Multi-broker - 30 Threads	Audit	16	20	46	46	37
		Comp.	17	23	73	73	54
	FRAMEWORK ΔT		30	38	109	109	82
Austrian Institute of Technology 1,000 Components	Single-broker - 15 Threads	Audit	1	1	1	1	1
		Comp.	1	1	1	1	1
	Multi-broker - 30 Threads	Audit	1	1	1	1	1
		Comp.	1	1	1	1	1
	FRAMEWORK ΔT		1	1	1	1	1

fragment the message over multiple batches. The 16,384 batch size was used to see how does the message count reflect on performance when only a part of JSON is sent in a message.

In Figure 8 and Figure 9 we outline the results of our measurements with respect to inbound data and record rate for the three above mentioned use cases. We aligned the color coding of the figures to highlight the correlation between the inbound data rate and number of records. The green lines shows the results of the most fundamental setup where on a single cluster 15 partitions were deployed within a single broker, blue lines show the results of the second setup where 5 brokers were configured on a single cluster with 80 working partitions uniformly deployed. Finally, the orange line represents the most advanced production configuration where we configured 3 clusters, each having again 5 brokers with total of 300 working partitions across clusters. In case of the small batch block (i.e., 4,096 bytes), we can see that regardless of the configuration setup message workload starts below 200 MB/sec, and depending on the configuration (i.e., config #1: 122 MB/sec - 662,918 records, config #2: 161 MB/sec - 874,875 records, config #3: 188 MB/sec - 1,016,145 records) differentiates the workload and record count, as shown by Figure 8 and Figure 9. All three configurations experience a slight drop in performance at the point of 50 security properties due to the batch block size (4,096 bytes) at the producer side that reaches the point where the producer can batch only single JSON message and therefore mitigates the performance growth slightly.

In contrast to the workload of the small batch block, the big batch block (i.e. 16,384 bytes) begins to process the workload at 200 MB/sec, and depending on the configuration (i.e., config #1: 203 MB/sec - 1,097,815 records, config #2: 240 MB/sec - 1,297,207 records, config #3: 250 MB/sec - 1,351,582 records) differentiates the workload and record count, as shown by Figure 8 and Figure 9. At the beginning from 1 SP to 10 SP, performance of multi broker configuration on single and multi-cluster setups are very close due to the small sizes of messages being transmitted. The multi-cluster configuration outperforms the remaining configurations by far, both in terms of workload and record rate, by reaching impressive 801 MB/sec. Furthermore, in case of 100 SP and 150 SP in Figure 8, we can see that the small batch outperforms the big batch due to its batch size.

Although, in both cases of small and big batch we reach very high inbound data rates, Figure 8, the record rate drops with the increase of SP per message due to the fact that the size of a JSON massage is increasing, Figure 9.

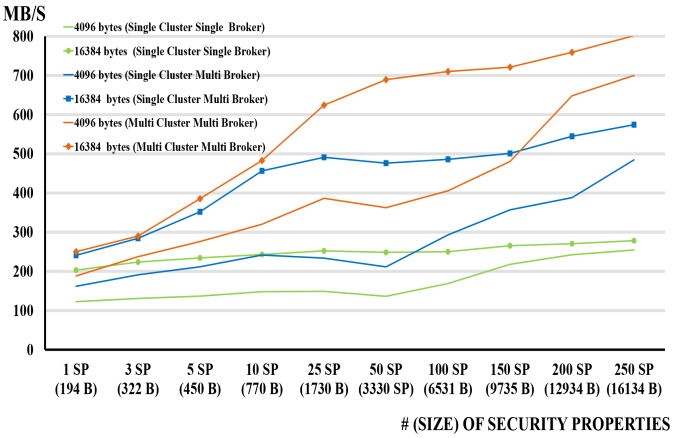

Figure 8: Simulation of inbound (consumption) data rate for our distributed messaging sub-system with various messages size that represents various security property count per message.

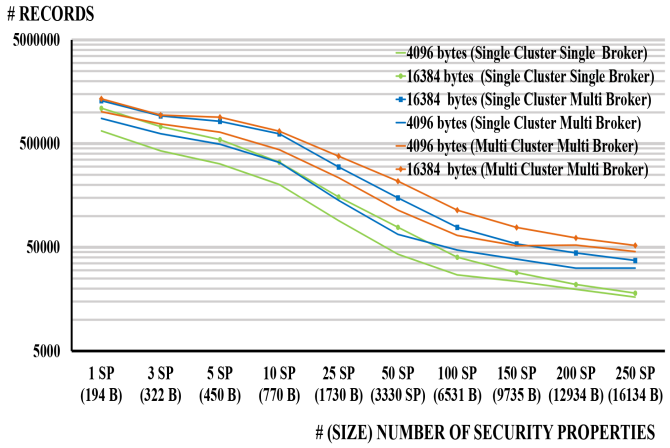

Figure 9: Simulation of inbound (consumption) record rate for our distributed messaging sub-system with various messages size with that represent different number of security property per message.

6.2.2. Processing Message Rate

The raw JSON messages after arrival to the messaging sub-system is subsequently consumed by the processing sub-system that simultaneously establishes an audit trail and conducts message transformation processes. In case of the audit trail messages are simply consumed and forwarded to the storage system, whereby in case of transformation they are being transformed into SPV that is used later to calculate assurance level. We depict the result of our assessment in Figure 10 for both auditing and transformation processes, whereby each of the processes was scaled with Apache Storm across 15, 30 and 80 threads. Since we deployed our processing sub-system in single c3.8xlarge instance the thread count is chosen based upon vCPU count approximately half and equal number of the vCPU number (i.e., 15 and 30 threads) to have some vCPU available for the system processes, and lastly the number equal to the number of partitions per broker, 80 threads. With respect to consumption we are limited to the number of consumption threads since per each partition it can be only one thread assigned to consume messages.

In case of the auditing process, we assign 15 threads to consume messages from the distributed messaging subsystem for single broker configuration, marked with full blue line in Figure 10. Next, we assigned 30 threads to consume messages from the multi-broker configuration, marked with full orange line in Figure 10. Furthermore, we investigated how the consumption performance of a single cluster is affected if we assign more than twice as much threads than available vCPUs (i.e., 32 vCPU). In Figure 10 the gray line highlights the performance for 80 consumption threads configuration. Based upon the results for 80 threads we concluded that if we increase the number of threads above the CPU count we would get more performance degradation since threads have to compete for the limited amount of available CPU resources. Hence, there was no need to test the third configuration setup with multi-clustered environment.

The complexity of performing tests for a scaled distributed processing environment, mandated that we first experimentally determined the overhead costs per SP by taking in to consideration 4 level depth general tree models and extrapolated the overhead costs with respect to audit process costs. The results shown in average 0.15% of overhead per individual security property. We used these SP overhead costs to extrapolate the assurance level calculation costs per security property with respect to audit process costs. The results of the estimations are marked with dashed blue and orange lines in Figure 10. The current limitations of our implementation shows that the consumption is starting to drop in both configuration setups after 50 SP since the costs to rise exponentially. This is an implementation detail challenge that we intend to address as a part of our future work.

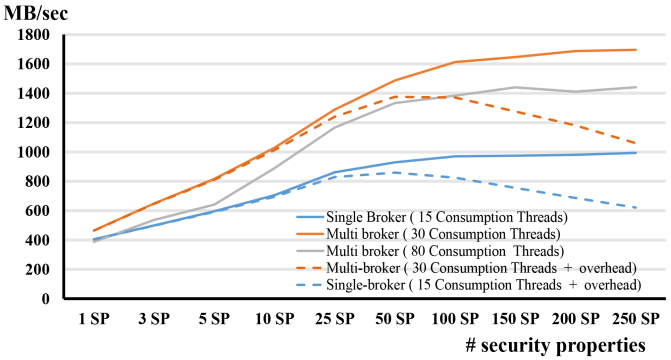

Figure 10: Results of consumption rate simulation for our distributed processing sub-system. We show the upper boundary of performance for single and multi broker setup with respect to different count of consumption threads.

6.3. Discussion

In this section, we discuss our findings based upon the evaluation results and identify shortcomings of our solution. We distinguish our result by highlighting the performance of the collecting cycle Δt of an individual scenario and general performance of our framework expressed as assessment cycle (ΔT), in Table 5. The Table 5 details the processing results of our system to perform auditing and computation of assurance level per for each use case scenario. Furthermore, we have only focused ourselves on the multi cluster configuration where we deployed single and multiple brokers, as the most efficient configuration for evaluating our results. Based upon the input workload, Figure 8, and processing results shown in Figure 10 we analyze the consumption rate (i.e., consumption processes scaled across 15 consumption threads or 30 consumption threads) affect the assessment cycle (ΔT) depending on the security property count in single and multi broker configurations shown by Figure 11, Figure 12, Figure 13, and Figure 14. In order to visualize better the collecting time

we intentionally left out the collecting cycle below 500,000 components for which Δt <4 sec.

In Figure 11 and Figure 12 we outline the performance results for single broker configuration setup with 15 consumption threads for auditing and transformation, respectively. The transformation overhead raises with the increase of SP per component take up to 37% more time (Δt_c = 124 seconds) with respect to auditing (Δt_a = 78 seconds) if we take into consideration the case with 5,000,000 components, to processes incoming workload, shown in Figure 12. In the second two figures, Figure 13 and Figure 14, we show the performance analysis results for multi broker configuration setup with 30 consumption threads. Regardless of the configuration setup the transformation overhead shows the same increase in percentage (up to 37% more, $\Delta t_c = 73$ seconds, time with respect to auditing, Δt_a = 46 seconds, process in case of 5,000,000 components). In case of multi-broker, Figure 14, we show that there is a significant increase in performance with respect to time required to perform both auditing and computation.

Next we compare the results for auditing and computing assurance level with respect to different configuration setup. When we take into consideration performance of auditing process, Figure 11 and Figure 13, where we have different number of consumption threads, 15 and 30 consumption threads (15 threads $\Delta t_a = 78$ seconds, and for 30 threads Δt_a = 46 seconds) respectively, our results show that by the increase of threads we get up to 42% increase in performance. In case when we compare the assurance computation overhead costs we get nearly the same performance boost in percentage, up to 41% (15 threads Δt_c = 124 seconds, and for 30 threads $\Delta t_c = 73$ seconds), but it unfortunately takes more time to process the messages.

We use the performance results depicted in Figure 13, and Figure 14, to evaluate the performance of real world use case scenarios shown in Table 5. As shown in Table 5 we distinguish our results per different cluster types (single or multi cluster), actions being taken (only audition the incoming messages or performing complete calculation and transformation) for different security related standards and best practices. We outline our evaluation by taking in to consideration our performance results for processing and messaging sub-systems and show the overall performance of our framework (ΔT). In case of Amazon cloud Frankfurt region scenario where we evaluate the performance across 750,000 components with 15 parallel threads, we show that our framework can perform auditing of 72 SP in a 4 seconds and 6 seconds in case of full assurance calculation. Furthermore, if we increase the SP count up to 240 it takes us 13 seconds to perform auditing and 19 seconds for assurance calculation. When we distributed the workload of across multiple clusters and increased the number of threads to 30, we noticed that the performance at the lover rate of SP did not change much whereby in case of higher count of SP, e.g. 240, we notice a significant improvement in performance. In case of the second cloud scenario, Amazon region Ireland, the analogy of performance has the same results, however with increased time interval required for processing due to the larger number of components. Our thirds scenario, the evaluation of an non-cloud environment Austrian institute of Technology infrastructure did not offer as much granularity in terms of results. Due to the relatively small number of components in our third scenario our framework manages to perform both auditing and computation of assurance level under 1 second.

Finally, we conclude our evaluation based up on the performance of our messaging system, Figure 8, and processing sub-system performance, Figure 10, we identify the throughput bottleneck of our framework caused by the current configuration of messaging sub-system and limitation in message production that under-performs in term of the inbound messages that can be fed in to our framework. Thus, our security assessment framework is currently limited to the messaging sub-system performance. However, we assume that in a real world scenario where the messages would be sent simultaneously from each component that would increase the incoming workload, and in that case we would be limited by the processing system performance. We therefore show for each of the evaluation scenarios in Table 5 under framework highlight the overall performance. In first evaluation scenario we see that our processing time Δt can slow down up to 50%, the second scenario due to the larger number of components which limits the processing shows up to 10% of performance. The non-cloud use case scenario due to the relatively small number of components doesn't show any significant changes in performance degradation.

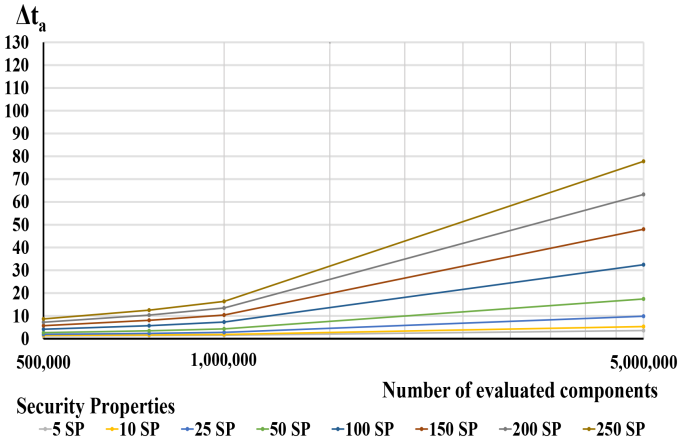

Figure 11: Message audit performance results in case of 15 consumption threads on a single broker setup. The results show the correlation between time required to audit information with respect number of components for different security property count.

7. Conclusion and Future Work

The acquisition of security related information across complex ICT environments is unfortunately ponderous,

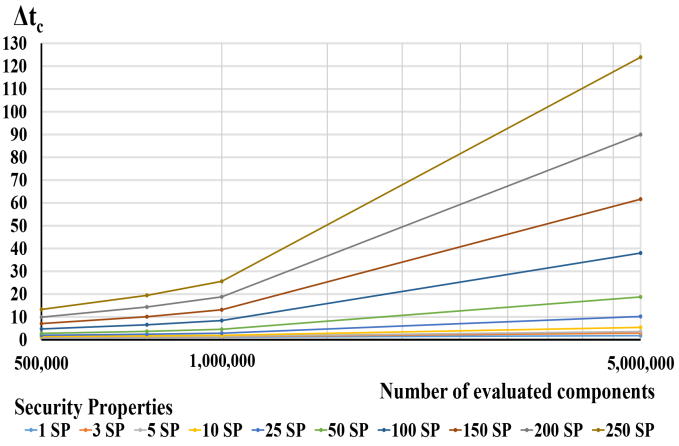

Figure 12: Assurance level computation performance results in case of 15 consumption threads on a single-broker setup. The results show the correlation between time required to audit information with respect number of components for different security property count.

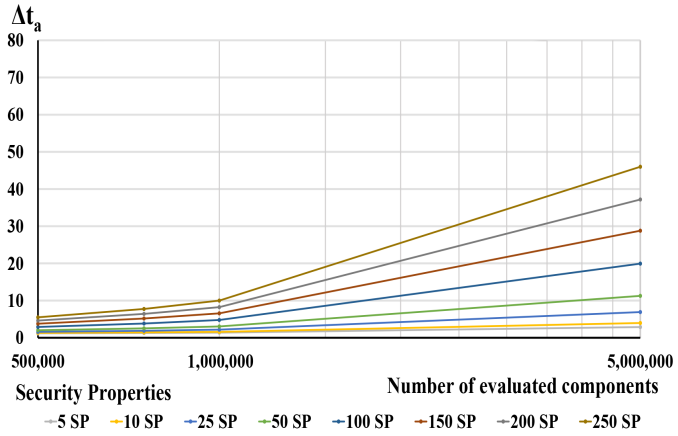

Figure 13: Message audit performance results in case of 30 consumption threads on a multi-broker setup. The results show the correlation between time required to audit information with respect number of components for different security property count.

time consuming process limited with human intervention, and therefore highly unpractical for large scale and volatile ICT environments. Furthermore, real time security assessment of hybrid cloud infrastructures that posses vast amount distinct components and include interdependencies is opposed by the countless restrictions that prevent sensitive information exposure. Motivated with these challenges we build our methodology for assessing security by including the relationship between individual layers and components owned by distinct, even competitive, stakeholders under the umbrella of single assessment domain. Therefore making it capable to assess highly complex and volatile systems such as hybrid clouds in nearly real time.

This work present a novel methodology for autonomic security assessment of complex multi-layered ICT infrastructures like cloud that can be owned by distinct stakeholders. We developed a concept for holistic service abstraction that is used as a blueprint during the security assessment for identifying interdependencies between components of a service or a system. Our solution can perform security analysis, even by a third party provider, and at the same time preserving the privacy of the system which service is being assessed. Furthermore, our solution provides a custom driven security assessment tailored according to the users needs and aligned with state of the art standards, best practices and guidelines. We also provide a detail experimental evaluation of our system where we stress individual sub-systems of our solution to identify potential bottlenecks.

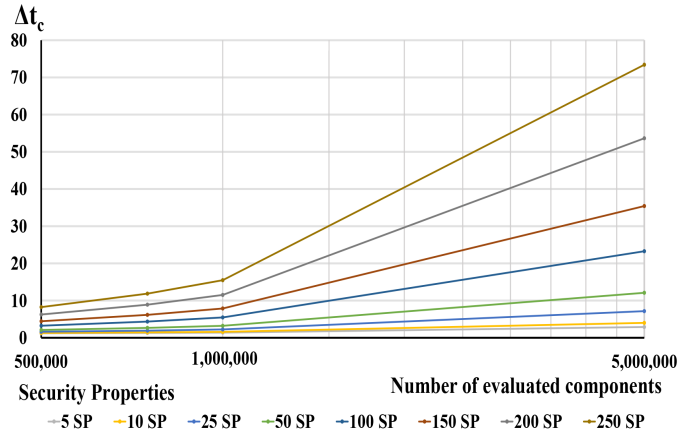

Figure 14: Assurance level computation performance results in case of 30 consumption threads on a multi-broker setup. The results show the correlation between time required to audit information with respect number of components for different security property count.

As a part of our future work we would like to perform additional optimization of our implementation and design to gain more performance. We also intend to test our system with various storage system modules like distributed file storage systems (Hadoop HDFS) and non-relational databases (e.g. MongoDB, Apache Cassandra). Furthermore, we intend to investigate how can our current solution can be put in commercial use to offer a Security as a Service module to the customers. Most importantly, we will focus our research on defining more efficient means, processes and concepts for gathering security related information that support our Security Assurance Assessment Framework. Recent studies indicate that digital forensic [68, 69] has taken significant momentum especially when it comes to Cloud environemnts [70, 71, 72, 73, 74]. Hence, due to its very nature to efficiently acquire digital evidence, we consider digital forensics will play an essential role in our future work.

Acknowledgments

This work has been supported and partially funded by the European Commission research projects SECCRIT *Grant No.* 312758, PRISMACLOUD *Grant No.* 644962 and CREDENTIAL *Grant No.* 653454.

Authors' biographies

Aleksandar Hudic is a Scientist in the Safety and Security Department of AIT, Austrian Institute of Technology. Aleksandar received his bachelors degree in Computer Engineering with emphasis on information system data analysis and masters degree in Information and Communication Technologies with emphasis on information system privacy and security, both obtained from the University of Zagreb, at Faculty of Electrical Engineering and Computing. He is currently enrolled in a PhD programme at Vienna University of Technology with the research topic focused mainly on the security and privacy of distributed systems.
Dr Paul Smith is a Senior Scientist in the Safety and Security Department of AIT, Austrian Institute of Technology. Previous to this appointment he was a Senior Research Associate at Lancaster University, UK, where he received his PhD in September 2003. Paul has been working in the area of network resilience for several years, publishing numerous conference and journal articles. Currently, he is coordinating the EU-funded SPARKS project, which is investigating the security and resilience of the smart grid his research focus in the project is on cybersecurity risk assessment.
Dr Edgar R. Weippl is research director of SBA Research and Associate Professor at the TU Wien. After graduating with a Ph.D. from the TU Wien, Edgar worked in a research startup for two years. He then spent one year teaching as an Assistant Professor at Beloit College, WI. From 2002 to 2004, while with the software vendor ISIS Papyrus, he worked as a consultant in New York, NY and Albany, NY, and in Frankfurt, Germany. In 2004 he joined the TU Wien and founded the research center SBA Research together with A Min Tjoa and Markus Klemen.

References

[1] S. M. Rinaldi, J. P. Peerenboom, T. K. Kelly, Identifying, understanding, and analyzing critical infrastructure interdependencies, IEEE Control Systems 21 (6) (2001) 11–25. doi:10.1109/37.969131.

[2] H. Takabi, J. B. D. Joshi, G. Ahn, Security and privacy challenges in cloud computing environments, IEEE Security & Privacy 8 (6) (2010) 24–31. doi:10.1109/MSP.2010.186.
URL http://doi.ieeecomputersociety.org/10.1109/MSP.2010.186

[3] A. Khajeh-Hosseini, D. Greenwood, I. Sommerville, Cloud migration: A case study of migrating an enterprise IT system to iaas, in: IEEE International Conference on Cloud Computing, CLOUD 2010, Miami, FL, USA, 5-10 July, 2010, 2010, pp. 450–457. doi:10.1109/CLOUD.2010.37.
URL http://dx.doi.org/10.1109/CLOUD.2010.37

[4] M. Jensen, J. Schwenk, N. Gruschka, L. L. Iacono, On technical security issues in cloud computing, in: IEEE International Conference on Cloud Computing, CLOUD 2009, Bangalore, India, 21-25 September, 2009, 2009, pp. 109–116. doi:10.1109/CLOUD.2009.60.
URL http://dx.doi.org/10.1109/CLOUD.2009.60

[5] N. H. A. Rahman, K. R. Choo, A survey of information security incident handling in the cloud, Computers & Security 49 (2015) 45–69. doi:10.1016/j.cose.2014.11.006.
URL http://dx.doi.org/10.1016/j.cose.2014.11.006

[6] N. V. Juliadotter, K. R. Choo, Cloud attack and risk assessment taxonomy, IEEE Cloud Computing 2 (1) (2015) 14–20. doi:10.1109/MCC.2015.2.
URL http://dx.doi.org/10.1109/MCC.2015.2

[7] S. Cimato, E. Damiani, F. Zavatarelli, R. Menicocci, Towards the certification of cloud services, in: IEEE Ninth World Congress on Services, SERVICES 2013, Santa Clara, CA, USA, June 28 - July 3, 2013, 2013, pp. 92–97. doi:10.1109/SERVICES.2013.16.
URL http://dx.doi.org/10.1109/SERVICES.2013.16

[8] M. Anisetti, C. A. Ardagna, E. Damiani, A low-cost security certification scheme for evolving services, in: 2012 IEEE 19th International Conference on Web Services, Honolulu, HI, USA, June 24-29, 2012, 2012, pp. 122–129. doi:10.1109/ICWS.2012.53.
URL http://dx.doi.org/10.1109/ICWS.2012.53

[9] M. Anisetti, C. A. Ardagna, E. Damiani, A certification-based trust model for autonomic cloud computing systems, in: Cloud and Autonomic Computing (ICCAC), 2014 International Conference on, IEEE, 2014, pp. 212–219.

[10] M. Anisetti, C. A. Ardagna, E. Damiani, A test-based incremental security certification scheme for cloud-based systems, in: 2015 IEEE International Conference on Services Computing, SCC 2015, New York City, NY, USA, June 27 - July 2, 2015, 2015, pp. 736–741. doi:10.1109/SCC.2015.104.
URL http://dx.doi.org/10.1109/SCC.2015.104

[11] M. Anisetti, C. A. Ardagna, E. Damiani, F. Gaudenzi, R. Veca, Toward security and performance certification of open stack, in: 8th IEEE International Conference on Cloud Computing, CLOUD 2015, New York City, NY, USA, June 27 - July 2, 2015, 2015, pp. 564–571. doi:10.1109/CLOUD.2015.81.
URL http://dx.doi.org/10.1109/CLOUD.2015.81

[12] M. Krotsiani, G. Spanoudakis, K. Mahbub, Incremental certification of cloud services, in: SECURWARE 2013-7th International Conference on Emerging Security Information, Systems and Technologies, 2013, pp. 72–80.

[13] M. Krotsiani, G. Spanoudakis, Continuous certification of non-repudiation in cloud storage services, in: 13th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, TrustCom 2014, Beijing, China, September 24-26, 2014, 2014, pp. 921–928. doi:10.1109/TrustCom.2014.122.

[14] S. Katopodis, G. Spanoudakis, K. Mahbub, Towards hybrid cloud service certification models, in: IEEE International Conference on Services Computing, SCC 2014, Anchorage, AK, USA, June 27 - July 2, 2014, 2014, pp. 394–399. doi:10.1109/SCC.2014.59.
URL http://dx.doi.org/10.1109/SCC.2014.59

[15] D. S. Herrmann, Using the Common Criteria for IT security evaluation, CRC Press, 2002.

[16] I. Drago, M. Mellia, M. M. Munafò, A. Sperotto, R. Sadre, A. Pras, Inside dropbox: understanding personal cloud storage services, in: Proceedings of the 12th ACM SIGCOMM Internet Measurement Conference, IMC '12, Boston, MA, USA, November 14-16, 2012, 2012, pp. 481–494. doi:10.1145/2398776.2398827.
URL http://doi.acm.org/10.1145/2398776.2398827

[17] C. Esposito, A. Castiglione, K. R. Choo, Encryption-based solution for data sovereignty in federated clouds, IEEE Cloud Computing 3 (1) (2016) 12–17. doi:10.1109/MCC.2016.18.
URL http://dx.doi.org/10.1109/MCC.2016.18

[18] I. O. for Standardization, I. E. Commission, Information technology security techniques evaluation criteria for it security - part 3: Security assurance components, Standard (July 2008).

[19] K. Rannenberg, Recent development in information technology security evaluation-the need for evaluation criteria for multilateral security., in: Security and control of information technology in society, 1993, pp. 113–128.

[20] I. O. for Standardization, I. E. Commission, Information technology security techniques security assessment of operational systems, Standard (July 2010).

[21] I. O. for Standardization, I. E. Commission, Information technology security techniques methodology for it security evaluation, Standard (July 2008).

[22] G. Spanoudakis, E. Damiani, A. Maña, Certifying services in cloud: The case for a hybrid, incremental and multi-layer approach, in: 14th International IEEE Symposium on High-Assurance Systems Engineering, HASE 2012, Omaha, NE, USA, October 25-27, 2012, 2012, pp. 175–176. doi:10.1109/HASE.2012.16.
URL http://dx.doi.org/10.1109/HASE.2012.16

[23] J. Such, A. Gouglidis, W. Knowles, G. Misra, A. Rashid, Information assurance techniques: perceived cost effectiveness, Computers and Security 60 (2016) 117–133. doi:10.1016/j.cose.2016.03.009.

[24] S. M. Khan, K. W. Hamlen, Computation certification as a service in the cloud, in: 13th IEEE/ACM International Symposium on Cluster, Cloud, and Grid Computing, CCGrid 2013, Delft, Netherlands, May 13-16, 2013, 2013, pp. 434–441. doi:10.1109/CCGrid.2013.75.
URL http://doi.ieeecomputersociety.org/10.1109/CCGrid.2013.75

[25] T. R. Groß, Efficient certification and zero-knowledge proofs of knowledge on infrastructure topology graphs, in: Proceedings of the 6th edition of the ACM Workshop on Cloud Computing Security, CCSW '14, Scottsdale, Arizona, USA, November 7, 2014, 2014, pp. 69–80. doi:10.1145/2664168.2664175.
URL http://doi.acm.org/10.1145/2664168.2664175

[26] S. Bleikertz, C. Vogel, T. Groß, Cloud radar: near real-time detection of security failures in dynamic virtualized infrastructures, in: Proceedings of the 30th Annual Computer Security Applications Conference, ACSAC 2014, New Orleans, LA, USA, December 8-12, 2014, 2014, pp. 26–35. doi:10.1145/2664243.2664274.
URL http://doi.acm.org/10.1145/2664243.2664274

[27] A. Sunyaev, S. Schneider, Cloud services certification, Commun. ACM 56 (2) (2013) 33–36. doi:10.1145/2408776.2408789.
URL http://doi.acm.org/10.1145/2408776.2408789

[28] I. Windhorst, A. Sunyaev, Dynamic certification of cloud services, in: 2013 International Conference on Availability, Reliability and Security, ARES 2013, Regensburg, Germany, September 2-6, 2013, 2013, pp. 412–417. doi:10.1109/ARES.2013.55.
URL http://dx.doi.org/10.1109/ARES.2013.55

[29] S. Schneider, J. Lansing, F. Gao, A. Sunyaev, A taxonomic perspective on certification schemes: Development of a taxonomy for cloud service certification criteria, in: 47th Hawaii International Conference on System Sciences, HICSS 2014, Waikoloa, HI, USA, January 6-9, 2014, 2014, pp. 4998–5007. doi:10.1109/HICSS.2014.614.
URL http://dx.doi.org/10.1109/HICSS.2014.614

[30] S. Lins, S. Thiebes, S. Schneider, A. Sunyaev, What is really going on at your cloud service provider? creating trustworthy certifications by continuous auditing, in: 48th Hawaii International Conference on System Sciences, HICSS 2015, Kauai, Hawaii, USA, January 5-8, 2015, 2015, pp. 5352–5361. doi:10.1109/HICSS.2015.629.
URL http://dx.doi.org/10.1109/HICSS.2015.629

[31] R. Savola, Towards a security metrics taxonomy for the information and communication technology industry, in: Software Engineering Advances, 2007. ICSEA 2007. International Conference on, 2007, pp. 60–60. doi:10.1109/ICSEA.2007.79.

[32] R. M. Savola, H. Abie, Development of security metrics for a distributed messaging system, in: Application of Information and Communication Technologies, 2009. AICT 2009. International Conference on, 2009, pp. 1–6. doi:10.1109/ICAICT.2009.5372566.

[33] R. M. Savola, P. Savolainen, A. Evesti, H. Abie, M. Sihvonen, Risk-driven security metrics development for an e-health iot application, in: 2015 Information Security for South Africa, ISSA 2015, Johannesburg, South Africa, August 12-13, 2015, 2015, pp. 1–6. doi:10.1109/ISSA.2015.7335061.
URL http://dx.doi.org/10.1109/ISSA.2015.7335061

[34] J. L. Garcia, H. Ghani, D. Germanus, N. Suri, A security metrics framework for the cloud, in: SECRYPT 2011 - Proceedings of the International Conference on Security and Cryptography, Seville, Spain, 18 - 21 July, 2011, SECRYPT is part of ICETE - The International Joint Conference on e-Business and Telecommunications, 2011, pp. 245–250.

[35] T. Heyman, R. Scandariato, C. Huygens, W. Joosen, Using security patterns to combine security metrics, in: Proceedings of the The Third International Conference on Availability, Reliability and Security, ARES 2008, March 4-7, 2008, Technical University of Catalonia, Barcelona , Spain, 2008, pp. 1156–1163. doi:10.1109/ARES.2008.54.
URL http://dx.doi.org/10.1109/ARES.2008.54

[36] R. Vaarandi, M. Pihelgas, Using security logs for collecting and reporting technical security metrics, in: Military Communications Conference (MILCOM), 2014 IEEE, 2014, pp. 294–299. doi:10.1109/MILCOM.2014.53.

[37] E. Caron, A. D. Le, A. Lefray, C. Toinard, Definition of security metrics for the cloud computing and security-aware virtual machine placement algorithms, in: 2013 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, CyberC 2013, Beijing, China, October 10-12, 2013, 2013, pp. 125–131. doi:10.1109/CyberC.2013.28.
URL http://dx.doi.org/10.1109/CyberC.2013.28

[38] I. Kotenko, O. Polubelova, I. Saenko, E. Doynikova, The ontology of metrics for security evaluation and decision support in siem systems, in: Availability, Reliability and Security (ARES), 2013 Eighth International Conference on, IEEE, 2013, pp. 638–645.

[39] K. Sun, S. Jajodia, J. Li, Y. Cheng, W. Tang, A. Singhal, Automatic security analysis using security metrics, in: MILITARY COMMUNICATIONS CONFERENCE, 2011-MILCOM 2011, IEEE, 2011, pp. 1207–1212.

[40] G. Aceto, A. Botta, W. de Donato, A. Pescapè, Cloud monitoring: A survey, Computer Networks 57 (9) (2013) 2093–2115. doi:10.1016/j.comnet.2013.04.001.
URL http://dx.doi.org/10.1016/j.comnet.2013.04.001

[41] K. Fatema, V. C. Emeakaroha, P. D. Healy, J. P. Morrison, T. Lynn, A survey of cloud monitoring tools: Taxonomy, capabilities and objectives, J. Parallel Distrib. Comput. 74 (10) (2014) 2918–2933. doi:10.1016/j.jpdc.2014.06.007.
URL http://dx.doi.org/10.1016/j.jpdc.2014.06.007

[42] T. A. B. Nguyen, M. Siebenhaar, R. Hans, R. Steinmetz, Role-based templates for cloud monitoring, in: Proceedings of the 7th IEEE/ACM International Conference on Utility and Cloud Computing, UCC 2014, London, United Kingdom, December 8-11, 2014, 2014, pp. 242–250. doi:10.1109/UCC.2014.33.
URL http://dx.doi.org/10.1109/UCC.2014.33

[43] G. da Cunha Rodrigues, G. L. dos Santos, V. T. Guimaraes, L. Z. Granville, L. M. R. Tarouco, An architecture to evaluate scalability, adaptability and accuracy in cloud monitoring systems, in: The International Conference on Information Networking 2014, ICOIN 2014, Phuket, Thailand, February 10-12, 2014, 2014, pp. 46–51. doi:10.1109/ICOIN.2014.6799663.
URL http://dx.doi.org/10.1109/ICOIN.2014.6799663

[44] R. Aversa, N. Panza, L. Tasquier, An agent-based platform for cloud applications performance monitoring, in: Ninth International Conference on Complex, Intelligent, and Software Intensive Systems, CISIS 2015, Santa Catarina, Brazil, July 8-10, 2015, 2015, pp. 535–540. doi:10.1109/CISIS.2015.79.
URL http://dx.doi.org/10.1109/CISIS.2015.79

[45] V. K. Naik, K. Beaty, N. Vogl, J. Sanchez, Workload monitoring in hybrid clouds, in: Cloud Computing (CLOUD), 2013 IEEE Sixth International Conference on, 2013, pp. 816–822. doi:10.1109/CLOUD.2013.145.

[46] V. Casola, A. D. Benedictis, M. Rak, Security monitoring in the cloud: An sla-based approach, in: 10th International Conference on Availability, Reliability and Security, ARES 2015, Toulouse, France, August 24-27, 2015, 2015, pp. 749–755. doi:10.1109/ARES.2015.74.
URL http://dx.doi.org/10.1109/ARES.2015.74

[47] R. van Renesse, K. P. Birman, W. Vogels, Astrolabe: A robust and scalable technology for distributed system monitoring, management, and data mining, ACM Trans. Comput. Syst. 21 (2) (2003) 164–206. doi:10.1145/762483.762485.
URL http://doi.acm.org/10.1145/762483.762485

[48] J. Shao, H. Wei, Q. Wang, H. Mei, A runtime model based monitoring approach for cloud, in: IEEE International Conference on Cloud Computing, CLOUD 2010, Miami, FL, USA, 5-10 July, 2010, 2010, pp. 313–320. doi:10.1109/CLOUD.2010.31.
URL http://dx.doi.org/10.1109/CLOUD.2010.31

[49] M. Rak, S. Venticinque, T. Máhr, G. Echevarria, G. Esnal, Cloud application monitoring: The mosaic approach, in: IEEE 3rd International Conference on Cloud Computing Technology and Science, CloudCom 2011, Athens, Greece, November 29 - December 1, 2011, 2011, pp. 758–763. doi:10.1109/CloudCom.2011.117.
URL http://dx.doi.org/10.1109/CloudCom.2011.117

[50] A. Brinkmann, C. Fiehe, A. Litvina, I. Lück, L. Nagel, K. Narayanan, F. Ostermair, W. Thronicke, Scalable monitoring system for clouds, in: IEEE/ACM 6th International Conference on Utility and Cloud Computing, UCC 2013, Dresden, Germany, December 9-12, 2013, 2013, pp. 351–356. doi:10.1109/UCC.2013.103.
URL http://dx.doi.org/10.1109/UCC.2013.103

[51] P. Massonet, S. Naqvi, C. Ponsard, J. Latanicki, B. Rochwerger, M. Villari, A monitoring and audit logging architecture for data location compliance in federated cloud infrastructures, in: 25th IEEE International Symposium on Parallel and Distributed Processing, IPDPS 2011, Anchorage, Alaska, USA, 16-20 May 2011 - Workshop Proceedings, 2011, pp. 1510–1517. doi:10.1109/IPDPS.2011.304.
URL http://dx.doi.org/10.1109/IPDPS.2011.304

[52] J. González, A. Muñoz, A. Maña, Multi-layer monitoring for cloud computing, in: 13th IEEE International Symposium on High-Assurance Systems Engineering, HASE 2011, Boca Raton, FL, USA, November 10-12, 2011, 2011, pp. 291–298. doi:10.1109/HASE.2011.65.
URL http://dx.doi.org/10.1109/HASE.2011.65

[53] A. Hudic, M. Tauber, T. Lorünser, M. Krotsiani, G. Spanoudakis, A. Mauthe, E. R. Weippl, A multi-layer and multitenant cloud assurance evaluation methodology, in: IEEE 6th International Conference on Cloud Computing Technology and Science, CloudCom 2014, Singapore, December 15-18, 2014, 2014, pp. 386–393. doi:10.1109/CloudCom.2014.85.
URL http://dx.doi.org/10.1109/CloudCom.2014.85

[54] M. Schöller, R. Bless, F. Pallas, J. Horneber, P. Smith, An architectural model for deploying critical infrastructure services in the cloud, in: IEEE 5th International Conference on Cloud Computing Technology and Science, CloudCom 2013, Bristol, United Kingdom, December 2-5, 2013, Volume 1, 2013, pp. 458–466. doi:10.1109/CloudCom.2013.67.
URL http://dx.doi.org/10.1109/CloudCom.2013.67

[55] R. E. Maeder, Computer Science with MATHEMATICA®: Theory and Practice for Science, Mathematics, and Engineering, Cambridge University Press, 2000.

[56] K. Scarfone, M. Souppaya, M. Sexton, et al., Guide to storage encryption technologies for end user devices, NIST Special Publication 800 (2007) 111.

[57] K. Scarfone, C. Tibbs, M. Sexton, et al., Guide to securing wimax wireless communications, NIST Special Publication 800 (2010) 127.

[58] B. Guttman, E. A. Roback, An introduction to computer security: the NIST handbook, DIANE Publishing, 1995.

[59] A. Auradkar, C. Botev, S. Das, D. D. Maagd, A. Feinberg, P. Ganti, L. Gao, B. Ghosh, K. Gopalakrishna, B. Harris, J. Koshy, K. Krawez, J. Kreps, S. Lu, S. Nagaraj, N. Narkhede, S. Pachev, I. Perisic, L. Qiao, T. Quiggle, J. Rao, B. Schulman, A. Sebastian, O. Seeliger, A. Silberstein, B. Shkolnik, C. Soman, R. Sumbaly, K. Surlaker, S. Topiwala, C. Tran, B. Varadarajan, J. Westerman, Z. White, D. Zhang, J. Zhang, Data infrastructure at linkedin, in: IEEE 28th International Conference on Data Engineering (ICDE 2012), Washington, DC, USA (Arlington, Virginia), 1-5 April, 2012, 2012, pp. 1370–1381. doi:10.1109/ICDE.2012.147.
URL http://dx.doi.org/10.1109/ICDE.2012.147

[60] G. Wang, J. Koshy, S. Subramanian, K. Paramasivam, M. Zadeh, N. Narkhede, J. Rao, J. Kreps, J. Stein, Building a replicated logging system with apache kafka, PVLDB 8 (12) (2015) 1654–1665.
URL http://www.vldb.org/pvldb/vol8/p1654-wang.pdf

[61] R. Ranjan, Streaming big data processing in datacenter clouds, IEEE Cloud Computing 1 (1) (2014) 78–83. doi:10.1109/MCC.2014.22.
URL http://doi.ieeecomputersociety.org/10.1109/MCC.2014.22

[62] A. Toshniwal, S. Taneja, A. Shukla, K. Ramasamy, J. M. Patel, S. Kulkarni, J. Jackson, K. Gade, M. Fu, J. Donham, N. Bhagat, S. Mittal, D. Ryaboy, Storm@twitter, in: Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data, SIGMOD '14, ACM, New York, NY, USA, 2014, pp. 147–156. doi:10.1145/2588555.2595641.
URL http://doi.acm.org/10.1145/2588555.2595641

[63] N. Santos, K. P. Gummadi, R. Rodrigues, Towards trusted cloud computing, in: Workshop on Hot Topics in Cloud Computing, HotCloud'09, San Diego, CA, USA, June 15, 2009, 2009.
URL https://www.usenix.org/conference/hotcloud-09/towards-trusted-cloud-computing

[64] I. O. for Standardization, I. E. Commission, Information technology – Security techniques – Information security management systems – Requirements, Standard (July 2005).
URL http://www.iso.org/iso/catalogue_detail?csnumber=42103

[65] Nist, E. Aroms, NIST Special Publication 800-53 Revision 3 Recommended Security Controls for Federal Information Systems and Organizations, CreateSpace, Paramount, CA, 2012.

[66] P. S. S. Council, Payment Card Industry Data Security Standard, https://www.pcisecuritystandards.org/, last accessed: 2016/03/25 (2016).

[67] OWASP, Application Security Verification Standard, https://www.owasp.org/, last accessed: 2016/03/25 (2016).

[68] N. H. A. Rahman, W. B. Glisson, Y. Yang, K. R. Choo, Forensic-by-design framework for cyber-physical cloud systems, IEEE Cloud Computing 3 (1) (2016) 50–59. doi:10.1109/MCC.2016.5.
URL http://dx.doi.org/10.1109/MCC.2016.5

[69] D. Quick, K. R. Choo, Forensic collection of cloud storage data: Does the act of collection result in changes to the data or its metadata?, Digital Investigation 10 (3) (2013) 266–277. doi:10.1016/j.diin.2013.07.001.
URL http://dx.doi.org/10.1016/j.diin.2013.07.001

[70] B. Martini, K. R. Choo, An integrated conceptual digital forensic framework for cloud computing, Digital Investigation 9 (2) (2012) 71–80. doi:10.1016/j.diin.2012.07.001.
URL http://dx.doi.org/10.1016/j.diin.2012.07.001

[71] B. Martini, K. R. Choo, Cloud storage forensics: owncloud as a case study, Digital Investigation 10 (4) (2013) 287–299. doi:10.1016/j.diin.2013.08.005.
URL http://dx.doi.org/10.1016/j.diin.2013.08.005

[72] D. Quick, K. R. Choo, Digital droplets: Microsoft skydrive forensic data remnants, Future Generation Comp. Syst. 29 (6) (2013) 1378–1394. doi:10.1016/j.future.2013.02.001.

URL http://dx.doi.org/10.1016/j.future.2013.02.001
[73] B. Martini, K. R. Choo, Distributed filesystem forensics: Xtreemfs as a case study, Digital Investigation 11 (4) (2014) 295–313. doi:10.1016/j.diin.2014.08.002.
URL http://dx.doi.org/10.1016/j.diin.2014.08.002
[74] D. Quick, K.-K. R. Choo, Google drive: Forensic analysis of data remnants, J. Netw. Comput. Appl. 40 (2014) 179–193. doi:10.1016/j.jnca.2013.09.016.
URL http://dx.doi.org/10.1016/j.jnca.2013.09.016

List of Figures