Towards Self-Protective Multi-Cloud Applications
MUSA – a Holistic Framework to Support the Security-Intelligent Lifecycle Management of Multi-Cloud Applications

Erkuden Rios[1], Eider Iturbe[1], Leire Orue-Echevarria[1], Massimiliano Rak[2] and Valentina Casola[3]
[1]*TECNALIA. ICT-European Software Institute. Parque Tecnológico de Bizkaia. C/ Geldo Edificio 700. E-48160 Derio (Spain)*
[2]*Dipartimento di Ingegneria Industriale e dell'Informazione, Seconda Università di Napoli, via Roma 29, Aversa (CE) (Italy)*
[3]*Dipartimento di Ingegneria Elettrica e delle Tecnologie dell'Informazione, da Università di Napoli, via claudio, Napoli (Italy)*
{erkuden.rios, eider.iturbe, leire.orue-echevarria }@tecnalia.com, massimiliano.rak@unina2.it, casolav@unina.it

Keywords: Multi-cloud, Security-by-design, Cloud SLAs, QoSec, Distributed Deployment, DevOps.

Abstract: The most challenging applications in heterogeneous cloud ecosystems are those that are able to maximise the benefits of the combination of the cloud resources in use: multi-cloud applications. They have to deal with the security of the individual components as well as with the overall application security including the communications and the data flow between the components. In this paper we present a novel approach currently in progress, the MUSA framework. The MUSA framework aims to support the security-intelligent lifecycle management of distributed applications over heterogeneous cloud resources. The framework includes security-by-design mechanisms to allow application self-protection at runtime, as well as methods and tools for the integrated security assurance in both the engineering and operation of multi-cloud applications. The MUSA framework leverages security-by-design, agile and DevOps approaches to enable the security-aware development and operation of multi-cloud applications.

1 INTRODUCTION

Cloud computing is an emerging promising paradigm for enabling new business models and economies of scale based on on-demand provisioning of IT resources (both hardware and software) over a network as metered services, where consumers are billed only for what they consume. A recent IDC Cloud forecast shows that the investment on public cloud services is expected to be more than €77,287 million in 2017 (IDC Cloud research, 2013).

Nevertheless, enterprises consider security as the #1 inhibitor to cloud adoptions (Waidner, 2009) (Expert Group Report. European Commission, 2010). Companies are reluctant to adopt cloud computing because of the difficulty in evaluating the trade-off between cloud benefits and the additional security risks and privacy issues it may bring. Most concerns are related to data protection, regulations compliance (Symantec, 2013) (Bitcurrent cloud computing survey, 2011) and other issues due to lack of insight (of controls and governance processes) in the outsourcing of data and applications: data confidentiality, trust on aggregators, control over data and/or code location, and resource assignment in multi-tenancy (Expert Group Report. European Commission, 2010). Businesses that want to exploit cloud computing need to be vigilant in understanding the potential privacy and security breaches in this new environment (Hubbard & Sutton, 2010).

Secure cloud environments are even more challenging today, since they are becoming more and more complex in reference to the number of *cloud resource types* that are available "as a service". Besides the traditional three service models defined by the NIST (Mell & Grance, 2010) (IaaS, PaaS and SaaS), new models are showing up such as Network as a Service specified by the ITU-T or Data as a Service defined in ISO/IEC 17826:2012 (ISO/IEC 17826:2012, 2012).

As the number of cloud models, cloud resources and cloud service providers grow in the market, it becomes theoretically easy (but not necessarily

technically) for the cloud consumer to deploy and use multiple cloud solutions at the same time in an integrated way (Miller, 2013). This means that despite the diverse characteristics of the cloud resources such as own management APIs and own service level offerings (both functional and security), all need to be monitored and managed as an integrated working entity.

The most challenging applications in heterogeneous cloud ecosystems are those that are able to maximise the benefits of the combination of the cloud resources in use: multi-cloud applications. For the context of this paper, a *multi-cloud application* is understood *as a distributed application over heterogeneous cloud resources whose components are deployed in different cloud service providers and still they all work in an integrated way and transparently for the end-user.*

Multi-cloud application solutions have to deal with the security of the individual components as well as with the overall application security including the communications and the data flow between the components. Even if each of the cloud service providers offered its own security controls, the multi-cloud application has to ensure an integrated security across the whole composition. Therefore, the overall security depends on the security properties of the application components, which in turn depend on the security properties offered by the cloud resources they exploit. For instance, the database component in charge of storing sensitive data cannot ensure a high confidentiality if the cloud storage resource in which it is deployed does not use strong encryption algorithms. Consequently, the whole multi-cloud application may be not sufficiently safe.

The paper is structured as follows. Section 2 describes the intended advances over the state of the art. Section 3 explains the MUSA approach and introduces the MUSA framework. Section 4 explains the future validation of the framework in industrial case studies. Finally, section 5 discusses the future work.

2 STATE OF THE ART

As outlined in introduction, the main purpose of the MUSA framework is to offer a solution to build security-aware multi-cloud applications. This research activity involves many different open research aspects, among them we focus on the following questions: *How do we identify the security requirements of a multi-cloud application? How can we ensure security of a multi-cloud application even when control over some of its components is not granted?? How do we deploy a multi-cloud application maintaining the promised security features?*

The following three subsections try to offer a brief summary of the state of the art of the existing replies to such questions.

2.1 Security-by-design in Multi-Cloud Applications

Security by design (SbD) was first positioned by Gartner (Kreizman & Robertson) and pointed out the importance of incorporating security into the enterprise architecture process since the beginning, i.e. by including security requirements in the design process. In addition, Gartner recently defined the Runtime Application Self-Protection (RASP) (Gartner) security concept which is a security technology capable of controlling the application execution and detecting and preventing real-time attacks. The concept behind this idea would be that the application itself is able to control and manage security mechanisms embedded in the application or which can be invoked as a service by the application.

Security Control frameworks are widely adopted tools used to identify the security controls required to ensure the protection of an ICT system. A security control is a safeguard or a countermeasure prescribed to protect a system and meet a set of defined security requirement. Control Frameworks are a structured list of security controls that help a security expert to select the checks to perform in order to guarantee the respect of security requirements of a given system. Example of such Control Frameworks are the NIST Control Framework (NIST 800-53r4, 2013) and the ISO/IEC 27001 (ISO/IEC 27001).

In Cloud environments such frameworks are of limited use since they mostly miss specific cloud related security controls. Nevertheless, several attempts to address these issues have been made in (NIST SP500, 2010), (Cloud Data Protection Cert, 2013), (Cloud Security Alliance, 2014).

2.2 Security Aware SLAs in Multi-Cloud Applications

As stated above, in this paper, multi-cloud applications are distributed applications that run consuming cloud resources. Such an approach implies that the multi-cloud application developer and owner (i.e. the one that runs it and offers its services) have no control over the real execution

environment of the application. This inhibits the correct evaluation of the security controls.

The approach almost universally followed to define guarantees for users of a service is the introduction of Service Level Agreements (SLAs). An SLA is a formal agreement between a service provider and its end user that describes functional and non-functional aspects of the provided target service, together with clearly defined responsibilities of the involved parties.

The most well-known machine-readable SLA models are the Open Grid Forum's Web Services Agreement (WS-Agreement) (Hubbard & Sutton, 2010) and IBM's Web Service Level Agreement (WSLA) (VukoliĆ, 2010). The WS-Agreement specification proposes a domain-independent and standard way to create SLAs while its predecessor WSLA seems to be deprecated.

SLAs appear as a successful method to guarantee common Quality of Service parameters, like availability and performance indicators. As stated in many recent works, such as (Kandukuri, Paturi, & Rakshit, 2009), in order to deal with security requirements in the Cloud ecosystem, SLAs should be actually used to define target service security parameters.

Security Service Level Agreements (often named SecLA), are recognized as a promising way to model security issues between Cloud Service Providers and their users. ENISA, in (Dekker & Hogben, 2011), has also identified the importance of SecLAs in the Cloud computing field, pointing out that, in many circumstances, customers are not aware of many acquired services security aspects.

As introduced in (Almorsy, Grundy, & Ibrahim, 2011) and in (Luna et al, 2013), the current dearth of reasoning techniques on Security SLAs is preventing the diffusion of these approaches in production environments. Nevertheless, currently, many efforts are being made to fill this gap. For example, in (Luna et al, 2013), authors aim to outline techniques to quantitatively reason about Cloud Security SLAs, defining security metrics and a proof of concept semi -automated framework in order to assess cloud security of different providers.

Several European projects have worked or are working in this subject focusing mainly on SecSLA negotiation (SPECS Project, 2014), the creation of a security-aware SLA based language and related cloud security dependency model (CUMULUS project) and on the accountability for cloud-based services (A4Cloud Project, 2014).

2.3 Security Driven Dynamic Deployment of Multi-Cloud Applications

Multi-cloud applications have complex composition, provisioning and deployment requirements, and the application design becomes even more complex at the time an additional aspect such as security enters in the equation. Therefore, several initiatives are running in order to support this type of activities.

CloudML (CloudML project, 2013) (Ferry et al, 2013) developed a domain-specific language to support the specification of provisioning, deployment and adaptation concerns related to multi-cloud systems at design-time and their enactment at runtime. CloudML's background is PIM4Cloud language, defined in REMICS project (REMICS Consortium, 2012) (Ferry, Chauve, Rossini, Morin, & Solberg, 2013).

Based on CloudML, different approaches (ARTIST Consortium, 2013) (ModaClouds consortium, 2013) (PaaSage Consortium, 2014) and versions of CloudML have been recently released to provide means to the design of cloud based applications deployment. In this context where there are multiple CloudML versions, a joint task force has been started by MODAClouds, PaaSage and ARTIST projects which goal is to define a unique common CloudML specification (ARTIST Consortium, 2013).

Another approach that can be followed includes TOSCA (OASIS, 2013). The TOSCA specification aims to enhance the portability of cloud applications and services by using a language for defining both the service components of distributed applications and the service management interfaces (Antonescu, Robinson, & Braun, 2012). This approach is currently being followed by SeaClouds (Seaclouds consortium, 2013).

3 MUSA APROACH: THE MUSA FRAMEWORK

Multi-cloud solutions represent a new challenging field in order to add value to overall cloud client experience (VukoliĆ, 2010). In order to exploit multi-clouds potentialities, different architectural approaches can be adopted (Bohli et al, 2013):

(i) replication of applications, i.e. the same system is deployed in more than one provider and malicious attacks can be easily discovered comparing operation results;
(ii) partition of application system into tiers, that allows to separate logic from data;

(iii) partition of application logic into fragments, that obfuscates the overall application logic to providers;
(iv) partition of application data into fragments, that makes impossible to a single provider to reconstruct data, safeguarding confidentiality.

MUSA aims at ensuring the security in all multi-cloud environments including those that combine multiple scenarios as described above. To this aim, MUSA approach combines i) a preventive security approach, promoting Security by Design practices in the development and embedding security mechanisms in the application, and ii) a reactive security approach, monitoring application runtime to mitigate security incidents, so multi-cloud application providers can be informed and react to them without losing end-user trust in the multi-cloud application.

In order to ensure the preventive oriented security to be embedded and aligned with reactive security measures, MUSA supports an integrated coordination of all phases in the application lifecycle management.

3.1 The MUSA Framework

The MUSA framework presented in this paper is intended to provide support the integration of the security within the multi-cloud application lifecycle, as illustrated in Figure 1. MUSA supports the first phase of the multi-cloud application lifecycle, the *development phase*, through the MUSA IDE, which helps in both specifying the end user security requirements and integrate such requirements in the application development.

The MUSA Decision support tool and MUSA Distributed deployment tool support the multi-cloud application *deployment phase*, helping in the choice of the cloud service provider and deployment of the multi-cloud application deployment.

The MUSA security assurance platform (SaaS) supports the last phase of the multi-cloud application lifecycle (*execution phase*), monitoring the application execution and, when needed, applying correction actions to grant the security features.

The MUSA framework aims to define a set of best practices and guidelines for the integrated management of Security by Design mechanisms in the lifecycle of multi-cloud secure applications, based on DevOps and agile (AgileManifesto, 2001) methodologies' principles. The practices are supported by the different automation tools provided in the MUSA framework, which enable the coordination between programming and deployment infrastructure worlds, ensuring the continuous alignment of multi-cloud application security requirements specification (both at composition and SLA levels), implementation, monitoring and enforcement.

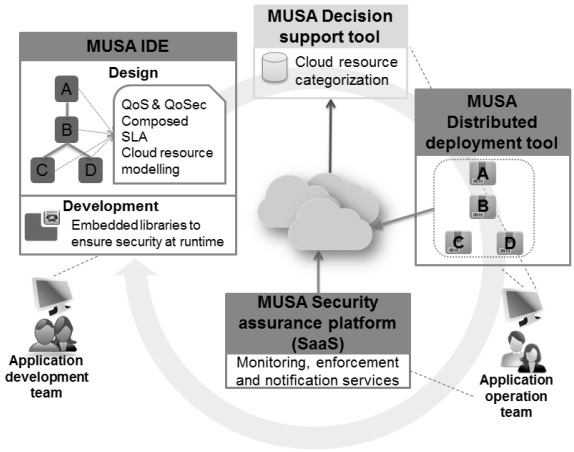

Figure 1: MUSA approach.

Following section describes the proposed tools in more detail.

3.1.1 Multi-cloud Secure Applications Design

For an effective design of multi-cloud secure applications an integrated development environment (IDE) is needed. To solve this requirement, MUSA framework intends to deliver an IDE that allows the design of application components taking into account security requirements. This IDE will be based on existing open-source solutions and will include three main modules.

The first module is a security requirements specification tool for multi-cloud applications, taking into consideration multi-cloud SLA definition and composition. The tool will allow expressing in the multi-cloud application SLA the security requirements (QoSec) together with functional and business requirements (QoS). To this aim, the needed components' and cloud resources' SLA composition shall be computed.

The second module supports the design of the breakdown of multi-cloud application into components based on the combination of *functional, business and security properties* that the multi-cloud application should offer. The tool will be based on existing standards such as CloudML (CloudML project, 2013) and TOSCA (OASIS, 2013), and will help application developers to design the architecture and the components composition taking into account the SLAs of the cloud resources in which the components will be deployed;

The last tool has the goal of design the provisioning and the deployment configuration of the needed heterogeneous cloud resources at multiple clouds layers (IaaS, PaaS).

Moreover, the MUSA IDE will include a set of security libraries that, embedded in the multi-cloud application, will enable the activation of security mechanisms and security controls without modifying the programming model. The security libraries aim at proposing a non-intrusive approach to introduce security in multi-cloud applications. These libraries will be inserted into the multi-cloud application components at design time and will be the responsible for ensuring the overall security at runtime. This software will include detection of non-compliant behaviour and enforcement mechanisms to be executed at runtime.

3.1.2 Multi-cloud Secure Applications Deployment

Once the application has been successfully designed, it has to be deployed. Deploying multi-cloud applications on distributed and heterogeneous resources encompasses several challenges that are not addressed currently in the existing tools. To solve this challenge, MUSA aims to provide a secure multi-cloud deployment tool that offers a distributed deployment service based on the dynamic selection of the cloud service providers (CSPs) that match with the application risk analysis, the subsequent security requirements as well as functional and business needs. In order to achieve it, the MUSA framework bases its offering mainly over three components. The first one is the cloud resource categorization of CSPs based on the measures of the security and functional properties at real time. The second component, a decision support tool, allows the selection of the cloud resources which combination is compliant with the security and functional requirements specified in the multi-cloud application composite SLA, after a previous simplified process of risk analysis. Finally, the third component allows an automated deployment of the multi-cloud secure application, distributing each of the application components' packages towards the matched cloud resource.

3.1.3 Multi-cloud Secure Applications Runtime

Monitoring multi-cloud applications at runtime involves collecting metrics of QoS and QoSec parameters of both the components of the application and the cloud resources provisioned. MUSA aims to provide a monitoring service capable of collecting such measurements by using standard APIs (if they are used by the cloud service providers (CSP)), cloud interoperability frameworks such as jclouds (Apache, 2012), or measures provided by MUSA security embedded libraries.

Whenever an incident occurs, MUSA sends alerts to notify the application provider about detected security relevant incidents. Moreover, MUSA send alerts when an application is in risk of not fulfilling its SLA, and some preventive action needs to be taken in order to keep security parameters well counterweighted with performance or within the margins specified in the SLA, e.g. a redeployment of application components across a different combination of cloud resources.

Finally, the enforcement service offered by MUSA ensures that the multi-cloud application respects the security requirements in its SLA.

All three services (monitoring, notification and enforcement) are delivered in the form of a security assurance platform, packaged as a SaaS product. The MUSA SaaS security assurance platform collaborates closely with the embedded libraries to enforce the security protection of the multi-cloud application user's data, through mechanisms such as authentication, authorisation, data encryption, data location assurance, etc. when the cloud resources used do not offer such mechanisms.

The MUSA SaaS will store monitored security parameters over the cloud resources and components, and manage the necessary notifications and alerts, so as the multi-cloud application provider can early react to possible security breaches. Contract verification processes will require a mapping between low-level resource metrics and high-level security parameters of the cloud services. This process will be done at runtime by MUSA SaaS, which will provide real-time assessment supported by complex processing of composed measures of low-level metrics.

4 MUSA FRAMEWORK VALIDATION

The economic viability, user acceptance and practical usability of the MUSA framework is expected to be validated through piloting the solution in realistic industry environments representing highly relevant services for the European economy: airline flight scheduling systems and urban smart mobility services.

In the following, we summarize the research challenges faced in both case studies.

4.1 Case Study A: Airline Flight Scheduling Multi-Cloud Application

For our first case study we have selected NetLine/Sched product by Lufthansa Systems to demonstrate how MUSA framework benefits the integrated security management of this application that exploits a number of heterogeneous cloud resources.

The product NetLine/Sched supports all aspects of flight schedule development and management.

In this case study, we are particularly interested in researching on how to:

(i) allow NetLine/Sched developer declare the options regarding data localisation (e.g. location country of the files), data retention and deletion, data integrity, confidentiality, access control and availability, etc. and make possible that such policies are embedded in the application specification.
(ii) enable security properties are embedded into the deployed application artefacts (security-by-design) for their continuous control at operation.
(iii) allow deployment into secure multi-cloud and multi-provider environments.
(iv) provide automated security assurance, supported by continuous monitoring, enforcement and notification mechanisms.
(v) keep the NetLine/Sched operator informed about the discrepancies and/or adapts to such requirements even in those cases that a change in the architecture or composition of the clouds underneath is needed.

4.2 Case Study B: Smart Mobility Multi-Cloud Application

Our second case study is an urban smart mobility multi-cloud application in Tampere city in Finland. Tampere Region has almost half a million inhabitants with a modal share of: 16% public transport, 27% pedestrians and cyclists, 57% private cars.

Tampere City Council has a number of services exposed to allow companies and individual developers to develop, test and productize own traffic applications using public data. The services can be publicly accessed via Intelligent Transport Systems and Services (ITS) platform (Wikipedia ITS, 2014), which includes the public transport services APIs, other traffic related APIs, traffic data, etc.

In this case study Tampere University (TUT) will take the role of many entrepreneur citizens and companies (generally SMEs) that create innovative applications by combining freely available open services and datasets in the Web to create business. The multi-cloud application by TUT aims at supporting the energy efficient and sustainable multi-modal transit of Tampere citizens when commuting from home to work and vice versa.

The major challenges for MUSA in this case study are the following:

(i) Enhance security capabilities of innovative services in transportation and public infrastructure in Tampere.
(ii) Enable entrepreneurs and citizens willing to develop innovative services based on IST Factory to be able to easily integrate security-intelligence into their applications through the use of MUSA IDE.
(iii) Empower operators of the multi-cloud applications that integrate IST Factory cloud-based services to ensure security of data storage and exchange at runtime through the use of MUSA assurance tools.
(iv) Allow evaluating new service multi-cloud deployment implications by checking service dependencies on other network and cloud resources.

5 FUTURE WORK

Application growth, rise in complexity and need for interoperability create market opportunity for cloud integrators and multi-cloud providers by offering new capabilities in the existing complex cloud landscape (North Bridge in partnership with GigaOM Research, 2013).

Taking profit of this opportunity window, MUSA aims at contributing to building up the innovation capacity and technology excellence of the European software and service industry by proposing a solution to master the security-intelligent lifecycle of multi-cloud applications based on novel DevOps and security-by-design approaches.

In this paper we have presented the MUSA framework whose main goal is to support the security-intelligent lifecycle management of multi-cloud applications. There are a number of major challenges in the path:

(i) Enable the security aware design of distributed applications over heterogeneous cloud resources.
(ii) Automatic discovery and decision support system of combinations of cloud services that best match the required balance between security and functional properties.

(iii) Security assurance though continuous monitoring and integrated methods in both engineering and operation of multi-cloud applications.

The MUSA project which will lead to the development and validation of MUSA framework was launched on January 2015 and will last 36 months. Future publications on the progress of the framework are expected both online (www.musa-project.eu) and in future papers.

ACKNOWLEDGEMENTS

The project leading to this paper has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 644429.

REFERENCES

A4Cloud Project. (2014). Accountability For Cloud and Other Future Internet Services. Retrieved from Accountability For Cloud and Other Future Internet Services.: www.a4cloud.eu/

AgileManifesto. (2001, February 17). Manifiesto for Agile Development. Retrieved December 8, 2013, from Manifiesto for Agile Development: http://agilemanifesto.org/

Almorsy, M., Grundy, J., & Ibrahim, A. S. (2011). Collaboration-based cloud computing security management framework. IEEE International Conference on Cloud Computing (CLOUD) (pp. 364-371). IEEE.

Antonescu, A.-F., Robinson, P., & Braun, T. (2012). Dynamic Topology Orchestration for Distributed Cloud-Based Applications. NCCA, (pp. 116 - 223).

Apache. (2012). Apache jclouds. Retrieved April 2014, from Apache jclouds: http://jclouds.apache.org/

ARTIST Consortium. (2012). ARTIST Project . Retrieved April 15th, 2014, from ARTIST Project: http://www.artist-pro-ject.eu/

ARTIST Consortium. (2013, September). Deliverable 7.2.1. Cloud services modelling and performance analysis framework. Retrieved April 2014, from Deliverable 7.2.1. Cloud services modelling and performance analysis framework: http://www.artist-project.eu/sites/default/files/D7.2.1%20Cloud%20services%20modeling%20and%20performance%20analysis%20framework_M12_30092013.pdf.

ARTIST Consortium. (2013, September). Deliverable D4.3.1 Dissemination report. Retrieved April 2014, from Deliverable D4.3.1 Dissemination report: http://www.artist-project.eu/sites/default/files/D4.3.1%20Dissemination%20report_M12_01102013.pdf.

Bitcurrent cloud computing survey. (2011). Bitcurrent cloud computing survey 2011. Bitcurrent cloud computing survey 2011.

Bohli, J. et al. (2013). Security and Privacy Enhancing Multi-Cloud Architectures.

Cloud Security Alliance. (2014). Cloud Controls Matrix. Retrieved April 2014, from Cloud Controls Matrix: https://cloudsecurityalliance.org/research/ccm.

Cloud Data Protection Cert. (2013). Cloud Data Protection Cert. Retrieved April 2014, from Cloud Data Protection Cert: http://clouddataprotection.org/cert.

CloudML project. (2013). Model-based provisioning and deployment of cloud based systems. CloudML project. Retrieved April 2014, from Model-based provisioning and deployment of cloud based systems. CloudML project.: http://cloudml.org.

CUMULUS project. (n.d.). Certification infrastrUcture for MUlti-Layer cloUd Services. Retrieved from Certification infrastrUcture for MUlti-Layer cloUd Services: http://cumulus-project.eu/

Dekker, M., & Hogben, G. (2011). Survey and analysis of security parameters in cloud SLAs across the European public sector. Retrieved April 2014, from Survey and analysis of security parameters in cloud SLAs across the European public sector: http://www.enisa.europa.eu/activities/Resilience-and-CIIP/cloud-computing/survey-and-analysis-of-security-parameters-in-cloud-slas-across-the-european-public-sector.

Expert Group Report. European Commission, I. S. (2010). The Future of Cloud Computing: Opportunities for European Cloud Computing Beyond 2010.

Ferry, N. et al. (2013). Towards model-driven provisioning, deployment, monitoring, and adaptation of multi-cloud systems. CLOUD 2013: IEEE 6th International Conference on Cloud Computing, (pp. 887-894).

Ferry, N., Chauve, F., Rossini, A., Morin, B., & Solberg, A. (2013). Managing multi-cloud systems with the CloudML framework. NordiCloud'13: 2nd Nordic Symposium on Cloud Computing & Internet Technologies. Oslo, Normay.

Gartner. (n.d.). Gartner IT Glossary - Runtime Application Self-Protection (RASP). Retrieved April 2014, from http://www.gartner.com/it-glossary/runtime-application-self-protection-rasp (Retrieved April 2014).

Hubbard, D., & Sutton, M. (2010). Top Threats to Cloud Computing V1. 0. Cloud Secuirty Alliance.

IDC Cloud research. (2013, September). IDC Cloud research. Retrieved March 2014, from IDC Cloud research: http://www.idc.com/getdoc.jsp?containerId=prUS24298013.

ISO/IEC 17826:2012. (2012). ISO/IEC 17826:2012 Information technology -- Cloud Data Management Interface (CDMI).

ISO/IEC 27001. (n.d.). ISO/IEC 27001 Information Technology – Security Techniques – Information Security management Systems – requirements.

Kandukuri, B., Paturi, V. R., & Rakshit, A. (2009). Cloud security issues. SCC'09. IEEE International Conference on Services Computing, 2009., (pp. 517-520).

Kreizman, G., & Robertson, B. (n.d.). Incorporating Security into the Enterprise Architecture Process. Retrieved April 2014, from Incorporating Security into the Enterprise Architecture Process: http://www.gartner.com/DisplayDocument?ref=g_search&id=488575.

Luna, J., et al. (2013). Negotiating and Brokering Cloud Resources based on Security Level Agreements. CLOSER 2013, (pp. 533-541).

Mell, P., & Grance, T. (2010). The NIST definition of cloud computing. In ACM (Ed.), Communications of the ACM, 53, no. 6, p. 50.

Miller, P. (2013, September). Sector RoadMap: Multicloud management in 2013.

ModaClouds consortium. (2013, September). Deliverable 4.2.1 MODACloudML development – Initial version. Retrieved April 2014, from Deliverable 4.2.1 MODACloudML development – Initial version: http://www.modaclouds.eu/wp-content/uploads/2012/09/MODAClouds_D4.2.1_MODACloudMLDevelopmentInitialVersion.pdf.

NIST 800-53r4. (2013). 291 NIST Security and Privacy Controls for Federal Information Systems and Organizations. Retrieved April 2014, from 291 NIST Security and Privacy Controls for Federal Information Systems and Organizations: nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-53r4.pdf.

NIST SP500. (2010). 291 NIST Cloud Computing Standards Roadmap. Retrieved April 2014, from 291 NIST Cloud Computing Standards Roadmap: http://www.nist.gov/itl/cloud/upload/NIST_SP-500-291_Version-2_2013_June18_FINAL.pdf.

North Bridge in partnership with GigaOM Research. (2013). The future of cloud computing, 3rd annual survey 2013. Retrieved March 2014, from The future of cloud computing, 3rd annual survey 2013: http://www.northbridge.com/2013-cloud-computing-survey.

OASIS. (2013). Topology and Orchestration Specification for Cloud Applications Standard. Retrieved April 2014, from TOSCA standard by OASIS: www.oasis-open.org/committees/tc_home.php?wg_abbrev=tosca.

PaaSage Consortium. (2014, April 30). Deliverable D2.1.2: CloudML Implementation Documentation (First version). Retrieved from Deliverable D2.1.2: CloudML Implementation Documentation (First version): http://www.paasage.eu/images/documents/paasage_d2.1.2_final.pdf.

REMICS Consortium. (2012). Deliverable 4.1 PIM4Cloud. Retrieved March 2014, from Deliverable 4.1 PIM4Cloud: http://www.remics.eu/system/files/REMICS_D4.1_V2.0_LowResolution.pdf.

Seaclouds consortium. (2013). Seaclouds project. Seamless adaptive multi-cloud management of service-based applications. Retrieved from Seaclouds project. Seamless adaptive multi-cloud management of service-based applications: http://www.seaclouds-project.eu/project.html.

SPECS Project. (2014). Secure Provisioning of Cloud Services based on SLA management. Retrieved from Secure Provisioning of Cloud Services based on SLA management: http://specs-project.eu/

Symantec. (2013). Choosing a Cloud Hosting Provider with Confidence. Retrieved April 2014, from Choosing a Cloud Hosting Provider with Confidence: http://www.itwhitepapers.com/content20287.

VukoliĆ, M. (2010). The Byzantine empire in the intercloud. 41(3), 105-111.

Waidner, M. (2009, November). Cloud computing and security. Lecture Univ. Stuttgart (November 2009). Retrieved from Cloud computing and security. Lecture Univ. Stuttgart (November 2009).

Wikipedia ITS. (2014). Intelligent Transport Systems and Services (ITS) Factory Wiki. Retrieved April 2014, from Intelligent Transport Systems and Services (ITS) Factory Wiki: http://wiki.itsfactory.fi/index.php/ITS_Factory_Developer_Wiki.