Security SLAs for Federated Cloud Services

Karin Bernsmed*, Martin Gilje Jaatun*, Per Håkon Meland* and Astrid Undheim†
*SINTEF ICT, Norway
Email: {karin.bernsmed,martin.g.jaatun,per.h.meland}@sintef.no
†Telenor Corporate Development, Norway
Email: astrid.undheim@telenor.com

***Abstract*—The federated Cloud paradigm aims to provide flexible and reliable services composed of a mixture of internal and external mini-clouds, but this heterogeneous nature is also fuelling the security concerns of the customers. To allay the fears and deal with the threats associated with outsourcing data and applications to the Cloud, new methods for security assurance are urgently needed. This paper presents current work on Cloud Security Service Level Agreements and our approach on how to manage this in the context of hybrid clouds. The purpose is to facilitate rapid service composition and agreements based on the necessary security requirements and establish trust between the customer and provider.**

We also show how this can be applied on a realistic case study related to a hybrid Unified Communication service.

***Keywords*-Cloud computing; security; QoS; SLAs; service composition; federated Cloud**

I. Introduction

Cloud computing services can already be found almost everywhere, offering all possible IT services imaginable in an on-demand and scalable manner. However, a perceived lack of security has been identified as one of the main show-stoppers for the adoption of Cloud services. There are actually very few new and unique security issues related to Cloud computing; most of them have been investigated and addressed in the traditional system and network security context. For instance, mechanisms for data protection, access control, trust delegation, mitigation of DDoS attacks and code verification are well known and have been (more or less) successfully applied to large-scale systems and distributed software. However, some characteristics of Cloud computing are fundamentally new, such as multi-tenancy and on-demand elasticity, and these introduce a new set of threats we should be aware of [1].

To put it simply, Cloud computing is outsourcing, and outsourcing implies bidirectional trust relationships. In the Cloud, the responsibility for implementing and maintaining efficient security mechanisms will be in the hands of the provider. To overcome the fear of the Cloud, these providers need to convince their customers that their data and applications will be properly secured. With emerging federated Cloud services composed of several other services from different providers, we have a situation that implies a chain of transitive trust. Assuring the customer that adequate security mechanisms exist and are correctly implemented throughout the whole and possibly non-transparent provider chain is therefore a major challenge.

To mitigate the security risks associated with the Cloud, existing security mechanisms and their effectiveness should be formalized in contracts. A Service Level Agreement (SLA) is a common way to specify the conditions under which a service is to be delivered. Today, a typical SLA for a Cloud service is specified between the customer and the end-market provider, usually limited to availability levels and credits/penalties. The absence of security aspects in SLAs combined with the lack of methods for making objective comparisons between different service offerings makes it virtually impossible for Cloud providers to offer trustworthy services to their customers when 3^{rd} party providers are involved. This paper outlines a framework for security SLAs for federated Cloud services. The purpose is twofold; to facilitate rapid service composition and agreements based on the necessary security requirements, and to establish trust between the customer and the provider(s). The proposed method is based on current state-of-the-art and experiences from a case study related to a Unified Communication (UC) service, considered by telecom operators to be highly relevant for federated clouds. The same framework has previously been applied to a CRM case study by Bernsmed et al. [2].

The paper is organized as follows. Section II describes the UC case study with an offered service composed of several sub-services. Section III outlines the current state-of-the-art in the SLA field and highlights some of the gaps we have identified. Section IV explains our work with managing a security SLA lifecycle related to a framework for security mechanisms for federated Cloud services. In section V we apply the methods and the framework to the case study. Section VI discusses our work and the perspectives of future research, before section VII concludes the paper.

II. Unified Communication as a Cloud Service

Unified Communication (UC) is commonly understood as the integration of real-time and non-real-time communication services, delivered to the customer in a uniform, integrated way. UC is often described as the merger of four modalities or communication areas, namely voice, messaging, presence and conferencing [3]:

978-0-7695-4485-4/11 $26.00 © 2011 IEEE
DOI 10.1109/ARES.2011.34

- Voice and Telephony: fixed, mobile and soft telephony
- Messaging: email, voice mail, unified messaging
- Presence and Instant Messaging (IM): presence and location information, IM[1]
- Conferencing: voice, video and web conferencing with two or more parties

UC is currently evolving from premises-based service delivery to a service delivered using a Cloud infrastructure. In the latter case, it is called UC-as-a-Service (UCaaS), which is the focus of this case study.

From the customers' point of view, the main advantage of UCaaS compared to the traditional premises-based UC is the pay-per use model, where customers only pay for the services they need and the actual duration they are used. This model is thus particularly interesting for SMEs who often seek to avoid large investments. A UCaaS solution will also offer the possibility to have different services available for different employees, as well as support for adding and removing services on-demand. For a telecom operator considering acting as the UCaaS provider, a viable option is to take the role of hosting the voice service (that can include both fixed, mobile and soft telephony) and use partnerships/agreements with external service providers for delivery of additional services, such as email and presence. The telecom operator may also sell additional services such as VPN (possibly with Quality of Service (QoS) guarantees) as part of the UC bundle.

Figure 1 illustrates such a case where various elements of a UCaaS is composed of several specialized Cloud services. Here the UCaaS provider offers a UC solution based on its own voice service combined with messaging, conferencing and presence services from external service providers.

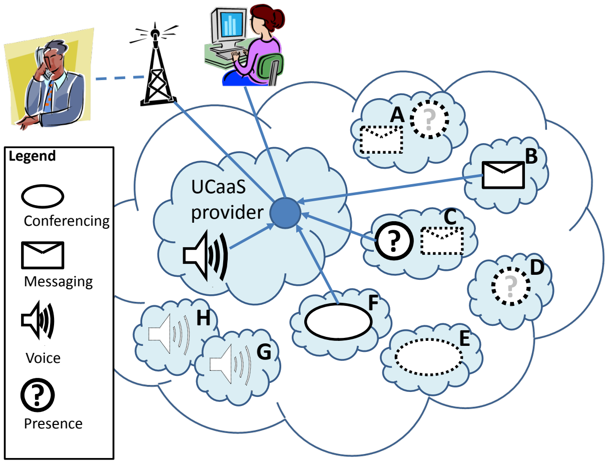

Figure 1. Unified Communication as a Service (UCaaS)

Federated Cloud services, such as the UCaaS illustrated in Figure 1, have been a vision for a long time, but adoption has been slow. This paper is devoted to one of the main reasons for this slow adoption - the lack of trust between the different parties involved in the service composition.

III. Cloud Security SLAs - State of the Art

Deploying services in the Cloud creates new challenges for both service providers and customers, especially regarding the service quality. The customers have less control of the service delivery, and need to take precautions in order not to suffer low performance, long downtimes or loss of critical data. Service Level Agreements (SLAs) have therefore become an important part of the Cloud service delivery model. An SLA is a binding agreement between the service provider and the service customer, used to specify the level of service to be delivered. This section provides a brief survey on the state-of-the-art of security in Cloud SLAs, spanning from existing industrial practices to ongoing research efforts.

A. SLAs in the Cloud

The technical part of the SLA (often called Service Level Specification (SLS)) includes information about the service delivered by the Cloud provider, the QoS provided, together with the penalties if the conditions are broken by the Cloud provider. The penalties are usually stated as service credits to the customers. In this context QoS refers to the (measurable) ability of a distributed system to provide network and computation services such that the customer's expectations are met.

In the Cloud, two apects of QoS have received special interest; *dependability* and *performance*. Service dependability is usually defined as a combination of the service availability (the proportion of time a system delivers service according to the requirements) and reliability (the ability to provide uninterrupted service), whereas performance is usually characterized by throughput (the number of bits per second of data transmitted or processed) and response time (the number of seconds needed for completing a specific task) [4]. The term QoS usually does not include security, even though some previous efforts have tried to extend the term in this respect [5], [6].

Today, most of the major Cloud service providers include QoS guarantees in their SLA proposals, however the focus in most cases is on dependability, measured as service availability[2]. In most cases, the SLA lacks performance guarantees, which from the customer's point of view is a major drawback. A very low performance will be perceived by the customer as service unavailability and should be credited accordingly.

[1] Strictly speaking, IM belongs in the messaging category, but historically, presence services were first offered by IM programs such as AIM and ICQ.

[2] An example is the Amazon EC2 Cloud service, which at the time of writing offers 99.95% availability on a yearly basis and issues 10% credits if the SLA is broken. Performance is not mentioned.

The on-demand characteristic of Cloud computing complicates the QoS provisioning and SLA management. The Cloud infrastructure needs to adjust to changing user demands, resource conditions and environmental issues. Hence, the Cloud management system needs to autonomically allocate resources to match the SLAs and also to detect possible violations and take appropriate action in order to avoid paying credits. Several challenges for autonomic SLA management still remain. First, resources need to be allocated according to a given SLA. Next, measurements and monitoring are needed to detect possible violations and react accordingly, e.g., by allocating more resources. Both of these actions require a mapping between low-level resource metrics and high-level SLA parameters. A proposal is given in [7] on how to adjust resource allocation to avoid SLA violations.

B. *Security SLAs*

Even though dependability and performance often are identified as critical issues, the number one barrier against adopting Cloud computing services is assurance [8]: how can potential customers be sure that it is safe to place their data and applications in the Cloud? Since the SLA is used to explicitly state the obligations of the provider, the implemented security mechanisms, their effectiveness, and the implications of possible mismanagement should be a part of this agreement. This concept is sometimes known as *Quality of Protection (QoP)*, which comprises the ability of a service provider to deliver service according to a set of specific security requirements. Security SLAs share the difficulties associated with the traditional QoS parameters, i.e., how to allocate appropriate resources according to security requirements and how to detect and react to possible violations.

There have been some projects in the research community looking into various aspects of security in SLAs. Early work on security agreements was performed in 1999 by Henning [9], who already then raised the question whether security can be adequately expressed in an SLA.

Security requirements for web services have been treated by Casola et al., who proposed a methodology to help evaluate and compare security SLAs [10]. Frankova and Yautsiukhin have also recognized the need for security in SLAs [11]. Their approach focuses on the process of selecting the optimal service composition based on a set of pre-defined requirements. Chaves et al. [12] explore security in SLAs applied on a monitoring and controlling architecture by Righi et al. [13]. As pointed out by Chaves et al., it is a challenge to define quantifiable security metrics, but they give examples related to password management, frequency of backups and repair/recovery time.

Organisations that use Cloud services can in some cases have strict policies on where their data should be stored, and according to Buyya et al. [14], it should be possible to have SLAs that specify the location of Cloud resources. A counter-example of this is given by Honeyball [15], who showed that it was not possible for EU customers of the Microsoft Azure platform to get any legally binding guarantee where their data would or would not be stored.

C. *Dynamic SLA Management*

SLA management today is mainly a static business where the contract terms are defined by the provider, typically published on a Web page, intended to be read by humans. With a shift towards a more dynamic service environment, where services change terms, are composed from resources in federated or hybrid clouds, and where more interactive SLA negotiations take place, the SLA management must become a more automatic process performed by software agents. More detailed SLAs would also be an argument for dynamic management as frequent service updates could easily cause more contract violations.

This paradigm has already been targeted in sister sciences to Cloud computing, such as Grid and Web services, but also here mostly in the context of traditional QoS (dependability and performance). It is therefore natural to take a look at the technology that enables SLA publishing, discovery and matching of contract templates, negotiation of terms, validation and monitoring.

SLAs for web services are composed of a set of Service Level Objectives (SLOs) that can be evaluated using measurable data called Service Level Indicators (SLIs). There are two main specifications that describe SLAs for web services. The first is the Web Service Agreement (WS-Agreement), developed by the Open Grid Forum (OGF) [16]. The second is the Web Service Level Agreement (WSLA) framework [17], which was developed by IBM for SLA monitoring and enforcement in a Service Oriented Architecture (SOA). The WSLA framework consists of an SLA definition language based on XML, and SLA management in terms of negotiation, monitoring and triggering of corrective actions when violations are detected.

Patel et al. [18] propose a mechanism for managing Cloud SLAs using the WSLA framework. Their main contribution is the usage of the 3^{rd} party support feature of WSLA to delegate the monitoring and enforcement part of the SLA management to trusted 3^{rd} parties. Relevant work has also been performed by Comuzzi et al. [19] and Theilmann et al. [20], contributing to reference architectures for multilevel SLA management.

There are also several ongoing research projects related to SLA management, such as the European projects SLA@SOI [21], ETICS [22] and mOSAIC [23].

D. *Open Issues*

Even though the above approaches to include security in SLAs are promising, they are designed for traditional service-oriented architectures. The federated Cloud paradigm, where services can be composed on demand and where the threat picture is somewhat different, warrants another look at the problem. Whereas several of the dynamic SLA management architectures proposed are adapted to the Cloud, they do not focus on security as an SLA attribute. The next section will present our approach to security SLAs for federated Cloud services, demonstrating its application on the case study presented in Section II.

IV. Security SLA Management for the Cloud

This section describes our proposed method for composing security SLAs for federated Cloud services. In this paper we focus on two parts of the proposed method; the lifecycle of a security SLA, and a framework for Cloud security mechanisms to include in the SLA.

A. Fundamental Concepts

The terminology we use is based on the framework architecture established in the SLA@SOI EU project [21], slightly adapted to fit into the Cloud security agreement context.

A *service* is a means of delivering value to customers. A service represents some function or type of task performed by a provider on behalf of a customer. Examples of services are a hotel booking service listed in a public registry or a computing service provided by Amazon. A *Service Provider (SP)* is an organization supplying services to one or more internal or external customers. An example is Amazon providing a Cloud computing service. A *Service Customer (SC)* (or just *customer*) is someone who orders/buys services. A *Service Level Agreement (SLA)* is an agreement between a service provider and a service customer. An SLA describes the service, documents service level targets, and specifies the responsibilities of the service provider and the customer. A single SLA may cover multiple services or multiple customers. A *security SLA* is an SLA that specifies the security obligations associated with a service. In contrast to traditional SLAs, a security SLA will include a set of *security requirements*. An SLA *offer* is a description of an agreement that a service provider is willing to accept. An SLA *template* (usually an XML document) can be used by either the service provider or the customer to advertise what types of offers they are willing to accept. In our model, security SLA templates and security SLA offers will be used by customers and service providers to state their requirements and capabilities regarding security in the SLA negotiation process.

B. The Security SLA Lifecycle

The security SLA lifecycle focuses on the interactions between the service customer and the service provider(s). Here we describe the necessary steps in the lifecycle. As can be seen in Figure 2, it consists of six different phases; publishing, negotiation, commitment, provisioning, monitoring and termination.

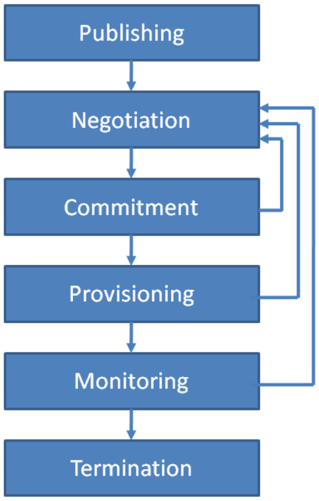

Figure 2. The security SLA lifecycle

1) Publishing: The first phase includes creating and publishing service offers in a service catalogue, in order to make them ready for discovery by potential customers. In this phase service providers will design and prepare security SLA templates based on their technical capabilities, business strategies and their relations with other service developers. This is a dynamic phase; service providers may frequently change their service offers, add new ones or withdraw templates with expired security services. A published security SLA template for a Cloud service must as a minimum contain the offered security mechanisms and their costs, and possibly also the penalty (customer credit) associated with breaking the agreement. In addition the validity period should be stated in a published SLA template.

2) Negotiation: In the negotiation phase the customer and the service provider agree on the details in the security SLA. This phase is further outlined in Figure 3. In our model the negotiation phase will consist of several steps. We do not delve into the details of the negotiation process here, but note that the resulting security SLA will be composed from a set of security requirements stated by the customer and the corresponding security offer from the service provider (who may also pose requirements to the customer). A negotiation can be performed by software agents acting on behalf of the customer and provider. As can be seen in the figure, the customer sends a list of security requirements to the service provider. In the hybrid Cloud context, the request may require resources that the service provider does not posess. The service provider will then match the stated requirements with the security SLA templates published in a service catalogue, and try to reserve resources from a number of other service providers who provide services that comply with the security requirements. The selection process can be either manual or partly automated.

The negotiation phase will result in security SLAs between the customer and the service provider, and possibly also between the service provider and other service providers. As can be seen in Figure 3 the final security SLA between the customer and the service provider can be a result of multiple interactions between a number of other providers.

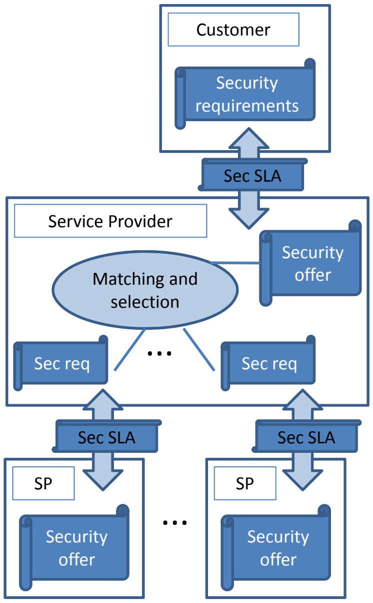

Figure 3. The basic concepts described in the Negotiation phase

3) Commitment: A successful negotiation phase will be followed by a commitment phase, where the security SLAs are digitally signed by all the involved partners. For a hybrid Cloud service the resulting security SLA between the customer and the service provider will usually be a result of multiple chains of contracts between the service provider and other providers.

4) Provisioning: The proposed lifecycle also includes a provisioning phase, to configure and accomplish the agreed security mechanisms. Change requests in either this phase or the commitment phase may lead to re-negotiation of security SLAs.

5) Monitoring: The monitoring phase is used to ensure that the details in the security SLAs are met. This phase includes detecting both past and ongoing security violations, and interpreting whether these events affect the agreed security SLAs. Detected violations of security SLAs may lead to either re-negotiating the SLAs or to termination of the service.

6) Termination: The termination phase ends the agreed contract and frees possibly reserved resources.

C. Security Requirements for the Cloud

A challenging part of the security SLA process lifecycle is to agree on what specific security mechanisms to include in the agreement. We have previously outlined a framework for security mechanisms in SLAs for Cloud services [2], illustrated in Figure 5. Here the security mechanisms are divided into three main service categories (**storage**, **processing** and **network**), depending on the particular Cloud resources that are used. The security mechanisms suggested for the framework have been categorized as described below:

Table I
SECURITY REQUIREMENTS FOR THE VOICE, MSG, PRESENCE AND CONF SERVICES

	Service	Id	Requirement	Priority
Secure Resource poling (RP)	VOICE, MSG, PRESENCE, CONF	RP3_1 (data encryption)	User profile information will be stored in an encrypted state	MUST
	VOICE, MSG, PRESENCE, CONF	RP3_2 (data encryption)	Only a hashed value of the user password will be stored.	MUST
	PRESENCE	RP2_1 (data deletion)	Presence information will not be permanently stored	SHOULD
	PRESENCE	AU4_1 (customer privacy)	It is the user's decision whether his online presence will be revealed to others	MUST
	VOICE, MSG, PRESENCE, CONF	AU4_2 (customer privacy)	A user profile will only requires a valid email address and username, Age, gender, name, picture and phone number will be optional fields.	NICE
	VOICE	RP13_1 (network encryption)	All audio streams will be sent over end-to-end encrypted channels	SHOULD
	VOICE, MSG, PRESENCE, CONF	RP14_1 (traffic isolation)	Information exchanged among the participants in a call will be kept confidential	MUST
	VOICE, MSG, PRESENCE, CONF	RP14_2 (traffic isolation)	Access to information exchanged among the participants in a call will be restricted to the participants	MUST
	MSG	RP15_1 (integrity protection)	All text messages will be digitally signed	SHOULD
Access Control (AC)	VOICE, MSG, PRESENCE	AC1_1 (identity management)	Authentication will be based on symmetric encryption using a trusted third party as authenticator	SHOULD
	VOICE, MSG, PRESENCE, CONF	AC1_2 (identity management)	The endpoints of all connections will be mutually authenticated	MUST
	VOICE, MSG, PRESENCE, CONF	AC2_1 (access management)	Only one instance of an authenticated user can participate in a communication session	NICE
Audit (AU)	VOICE, MSG, PRESENCE	AU1_1 (logging)	Only the service provider will have access to statistical information	SHOULD
	MSG	AU1_2 (logging)	Asymmetric communication will be stored in an encrypted state and not for more than 48 hours	SHOULD
	VOICE, MSG, PRESENCE, CONF	AU1_3 (logging)	All location data will be logged for a minimum of 48 hours and maximum of 168 hours.	MUST
Incident m. (IM)	VOICE, MSG, PRESENCE	IM5_1 (DoS mitigation)	Both client and servers will be protected against DoS attacks	NICE

- Secure Resource Pooling (RP): includes mechanisms for secure sharing of physical resources.
- Secure Elasticity (E): includes mechanisms that ensure secure migration of customer data and applications.
- Access Control (AC): includes mechanisms that protect the customer data from unauthorized access.
- Audit, Verification & Compliance (AU): includes mechanisms that facilitate auditing and verifying security properties, together with privacy preserving measures.
- Incident Management & Response (IM): includes mechanisms for detecting and responding to threats and unwanted events.

The purpose of the framework is to facilitate easier service selection and composition in the security SLA lifecycle.

The basic idea is to provide a set of standardized security mechanisms that a potential Cloud customer can choose between when outsourcing critical services to the federated Cloud.

V. Application on the UCaaS Case Study

We will now show how to apply the SLA lifecycle management from the previous section on a UC solution composed by a VoIP service (VOICE), a message service (MSG), a presence information service (PRESENCE) and a conferencing service (CONF) (see Figure 1).

In the first phase of the security SLA lifecycle (Figure 2) various providers publish their security offerings in accordance with the type of service offered. The customer will then approach the service provider with a request for a security service level in a prioritized list. In the UCaaS case study this list may contain the mechanisms and requirements outlined in Table I. As can be seen in the table, the priorities are given as three levels: MUST (the customer will not accept any service that does not fulfil this requirement), SHOULD (the customer would prefer this, but may settle for less) and NICE (nice to have, the customer may use it as a tie-breaker in otherwise equal offerings). The list of security requirements has been compiled using the Cloud SLA security mechanisms framework [2] and previous work on VoIP security [24], [25].

Based on the published security SLA templates, the UCaaS provider will then match the customer's security requirements with the possible service offers and make a short-list of all the providers who can comply with all MUST requirements. In this specific example, the UCaaS provider has its own voice service, and as long as this can satisfy all MUST voice-related requirements, the UCaaS provider will not query additional external providers. We will assume that this is the case for our example. For the remaining components, the UCaaS provider will initiate a negotiation for each sub-service, selecting relevant requirements from the list in each case. The message flow in the negotiation phase is outlined in Figure 4.

In the UCaaS implementation illustrated in Figure 1, one can see that messaging services are offered by service provider A, B and C. However, the matching process reveals that only provider B and C satisfy all the MUST priorities stated by the customer. The UCaaS provider will therefore request messaging services from these two only. Now assume that both providers respond that they can fulfil all the MUST and SHOULD requirements, however, provider B can in addition fulfil the two NICE requirements. Based on this, the UCaaS provider selects provider B to deliver the messaging service. The UCaaS provider then requests a presence service from provider C and D. Again it is easy for the UCaaS provider to choose provider C, since it turns out that it can fulfil all MUST and SHOULD requirements, while provider D can only fulfil the MUST requirements.

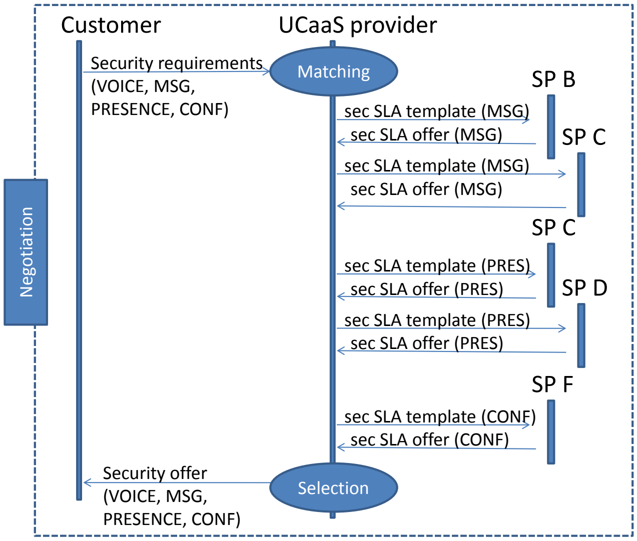

Figure 4. Message flow in the Negotiation phase

Finally, the UCaaS provider requests a conferencing service from provider F who responds with an offer containing all the MUST security requirements, but none of the SHOULD and NICE requirements. Having no other alternatives, the UCaaS provider chooses provider F to deliver the conferencing service.

After the negotiation phase is completed, the UCaaS provider presents the negotiated combined offer to the customer, who we may assume accepts this offer. The UCaaS provider then confirms the booking of the services from provider B, C and F; all of whom commit to provide the previously offered service (if any of the providers for some reason should fail to commit at this stage, the process returns to negotiation).

VI. Discussion and Further Work

Security in hybrid Clouds is quite a challenge, and we consider our work to still be in an early stage. In the UCaaS case study we have a situation where all the functional requirements are always satisifed by the sub-services. This is seldom the case in real life, but we have chosen not to focus our attention on compatibility and interoperability issues here. Likewise, security requirements might come in conflict with other QoS requirements in a SLA negotiation phase, and finding a balance that ensures a secure *enough* service with reasonable costs requires a more holistic lifecycle approach. Still, even within our focus area there are many issues left to be solved; some of the most urgent ones are:

- Security is inherently difficult to measure. Even for a Cloud service that has been in operation for some time it can be difficult to determine how secure it really is. In addition, the really clever attacks often go unnoticed.

- Security requirements are difficult to define, and are often stated as something that should *not* happen, making them difficult to use as input to an SLA template.
- Monitoring and controlling a security SLA is inherently difficult. The nature of clouds is to hide the inner workings from the customer.
- We need standardized, machine readable formats for the security SLA templates used in the publishing and negotiation phases, so that the roles can be played by software agents rather than humans.
- The combination of two or more secure services in a hybrid Cloud does not necessarily mean that the result is secure.

We are currently in the process of setting up a lab environment to implement and test the different steps in the proposed security SLA management lifecycle. The purpose is to evaluate to what degree existing standards and protocols can be used to realize the proposed architecture, and to determine the severity of a selected set of relevant attacks. In addition we will perform more research on *trust-based service selection* [26]; a concept that has received little attention so far, but that we believe will be a crucial part of Cloud service composition.

VII. Conclusion

Cloud federation brings together different service providers and their offered services so that many Cloud variants can be tailored to match different sets of customer requirements. To mitigate security risks and convince hesitant customers, security SLAs must be a part of the federated Cloud. In this paper we have described how to manage the security SLA lifecycle with the aid of a framework for security mechanisms as input to contract requirements. The purpose is to facilitate rapid service compositions and agreements for hybrid Clouds. There are still many unresolved issues related to security SLAs that need attention, and we are fully aware that security is something that cannot be handled in isolation from other requirements. Therefore, further work and practical experiences are needed within this research field.

Acknowledgment

This research has been supported by Telenor through the SINTEF-Telenor research agreement.

References

[1] Y. Chen, V. Paxson, and R. Katz, "What's New About Cloud Computing Security?" *University of California Berkeley Report No UCBEECS20105 January*, vol. 20, no. 2010, 2010. [Online]. Available: http://www.eecs.berkeley.edu/Pubs/TechRpts/2010/EECS-2010-5.pdf

[2] K. Bernsmed, M. G. Jaatun, and A. Undheim, "Security in Service Level Agreements for Cloud Computing," in *Proceedings of the 1st International Conference on Cloud Computing and Services Science, (CLOSER)*, 2011.

[3] B. Elliot and S. Blood, "Magic Quadrant for Unified Communications," 2009. [Online]. Available: http://www.gartner.com/DisplayDocument?doc_cd=201349

[4] International Telecommunication Union, "Terms and Definitions Related to Quality of Service and Network Performance Including Dependability, ITUT E.800," 2008.

[5] C. Irvine, "Quality of security service," in *Proc. ACM New Security Paradigms Workshop*, 2000, pp. 91–99.

[6] S. Lindskog, "Modeling and tuning security from a quality of service perspective," Ph.D. dissertation, Chalmers University of Technology, 2005.

[7] I. Brandic, V. C. Emeakaroha, M. Maurer, S. Dustdar, S. Acs, A. Kertesz, and G. Kecskemeti, "LAYSI: A Layered Approach for SLA-Violation Propagation in Self-managebale Cloud Infrastructures," in *Proceeding of the 2010 34th Annual IEEE Computer Software and Applications Conference Workshops*, no. i. Ieee, Jul. 2010, pp. 365–370.

[8] European Network and Information Security Agency (ENISA), "Cloud Computing: Benefits, risks and recommendations for information security," 2009.

[9] R. R. Henning, "Security service level agreements: quantifiable security for the enterprise?" in *Proceedings of the 1999 workshop on New security paradigms*, ser. NSPW '99. New York, NY, USA: ACM, 2000, pp. 54–60.

[10] V. Casola, A. Mazzeo, N. Mazzocca, and M. Rak, "A SLA evaluation methodology in Service Oriented Architectures," in *Quality of Protection*, ser. Advances in Information Security, D. Gollmann, F. Massacci, and A. Yautsiukhin, Eds. Springer US, 2006, vol. 23, pp. 119–130.

[11] G. Frankova and A. Yautsiukhin, "Service and protection level agreements for business processes," in *Young Researchers Workshop on Service*, 2007.

[12] S. A. de Chaves, C. B. Westphall, and F. R. Lamin, "SLA Perspective in Security Management for Cloud Computing," in *Proceeding of the 2010 Sixth International Conference on Networking and Services*. IEEE, March 2010, pp. 212–217.

[13] R. R. Righi, D. L. Kreutz, and C. B. Westphall, "Sec-mon: An architecture for monitoring and controlling security service level agreements," in *XI Workshop on Managing and Operating Networks and Services*, 2006.

[14] R. Buyya, C. S. Yeo, and S. Venugopal, "Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities," in *High Performance Computing and Communications, 2008. HPCC '08. 10th IEEE International Conference on*, sept. 2008, pp. 5 –13.

[15] J. Honeyball, "The truth about microsoft azure - and where your data will be kept," *PC Pro*, August 2009.

[16] OGF, "Web Services Agreement Specification (WS-Agreement)," Tech. Rep., 2007.

[17] H. Ludwig, A. Keller, A. Dan, R. P. King, and R. Franck, "Web Service Level Agreement (WSLA) Language Specification," IBM, Tech. Rep., 2003.

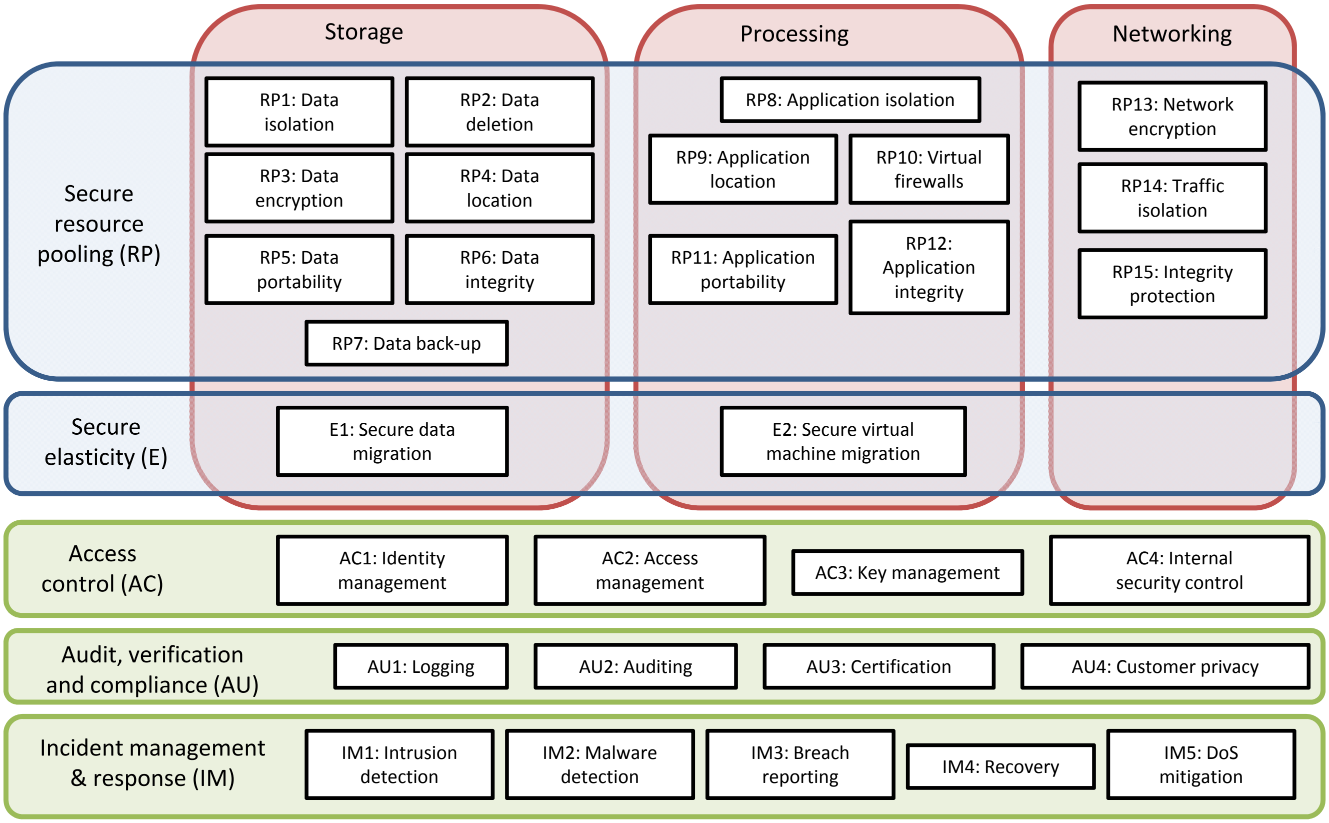

Figure 5. A framework for security mechanisms for Cloud SLAs [2]

[18] P. Patel, A. Ranabahu, and A. Sheth, "'Service Level Agreement in Cloud Computing," in *Proceedings of OOPSLA 2009)*, 2009.

[19] M. Comuzzi, C. Kotsokalis, C. Rathfelder, W. Theilmann, U. Winkler, and G. Zacco, "A framework for multi-level sla management," in *Proceedings of the 2009 international conference on Service-oriented computing*, ser. ICSOC/ServiceWave'09. Springer-Verlag, 2009, pp. 187–196.

[20] W. Theilmann, J. Happe, C. Kotsokalis, A. Edmonds, K. Kearney, and J. Lambea, "A Reference Architecture for Multi-Level SLA Management," *Journal of Internet Engineering*, vol. 4, no. 1, pp. 289–298, 2010.

[21] "SLA@SOI," 2011. [Online]. Available: http://sla-at-soi.eu/

[22] "ETICS (Economics and Technologies for Inter-Carrier Services)," 2011. [Online]. Available: https://www.ict-etics.eu/

[23] "mOSAIC (Open source API and platform for multiple clouds)," 2011. [Online]. Available: http://www.mosaic-cloud.eu/

[24] M. Marjalaakso, "Security requirements and constraints of VoIP," Department of Electrical Engineering and Telecommunications, Helsinki University of Technology, Tech. Rep., 2000. [Online]. Available: http://www.tml.tkk.fi/Opinnot/Tik-110.501/2000/papers/marjalaakso/voip.html

[25] D. R. Kuhn, T. Walsh, and S. Fries, "Security considerations for Voice Over IP systems," NIST, Tech. Rep. SP 800-58, January 2005.

[26] "Aniketos: Ensuring Trustworthiness and Security in Service Composition," 2011. [Online]. Available: http://www.aniketos.eu/