In: Multidisciplinary Research and Practice for Information Systems,
Volume 7465 of the series Lecture Notes in Computer Science, pp 123-130,
http://link.springer.com/chapter/10.1007/978-3-642-32498-7_10, 2012

Security SLAs – an idea whose time has come?

Martin Gilje Jaatun[1], Karin Bernsmed[1], and Astrid Undheim[2]

[1] Department of Software Engineering, Safety and Security
SINTEF ICT
NO-7465 Trondheim, Norway
http://www.sintef.no/ses
[2] Telenor Research and Future Studies
Trondheim, Norway

Abstract. Service Level Agreements (SLAs) have been used for decades to regulate aspects such as throughput, delay and response times of services in various outsourcing scenarios. However, security aspects have typically been neglected in SLAs. In this paper we argue that security SLAs will be necessary for future Internet services, and provide examples of how this will work in practice.

1 Introduction

The future Internet will provide an open ecosystem of services that can be mixed and matched according to customers' individual requirements. Service oriented architectures will form the basis for future applications and software products where complex service compositions and dynamic changes and replacement of individual service components will be common practice. New components will be picked not only based on functionality, but also based on their availability, performance, security, quality and price. In order to differentiate themselves in a highly competitive market, service providers will have to front the differentiating advantages of their services in order to attract potential customers.

The obvious downside of complex applications involving multiple providers is that they introduce the specter of uncertainty; it will be difficult for the customers to ensure that the final service compositions, i.e., the products they are paying for, behave as expected and that the individual service components can be trusted. From the customer's point of view, security and trustworthiness through the whole chain of service components may very well be the key issue that differentiates one potential service provider from another.

To illustrate the complexities of a composite service, we will employ the example depicted in Figure 1 [1]. Here, the service provider is a telecom operator with its own voice telephony service who wants to offer a Unified Communications (UC) service to its customers. To do this, it needs to combine its own voice service with conferencing, messaging and presence services from subcontractors. The telecom operator will hence act as a service provider towards the final UC users and as a service customer towards its subcontractors. In the following we will see how the UC provider can offer a defined security level to its customers through the use of Security Level Agreements with its subcontractors.

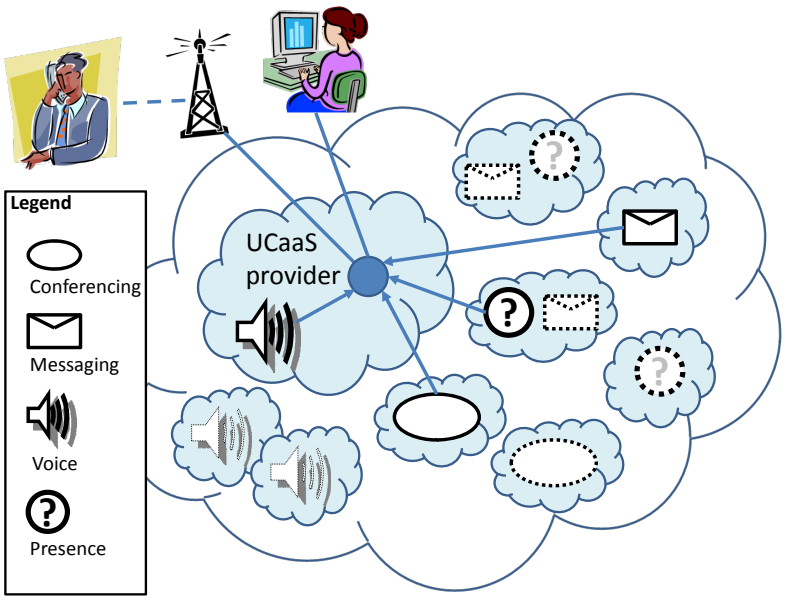

Fig. 1. Example of Unified Communication as a Composite Service[1]

2 Service Level Agreements

A Service Level Agreement (SLA) is a common way to specify the conditions under which a service is delivered. An SLA addresses three fundamental questions: what is delivered; where is it delivered; when is it delivered? The purpose of an SLA is to serve as a binding agreement between the service customer and the service provider. The SLA will help ensure that the service keeps the right level of quality and that customers are credited accordingly in terms of contract violations. SLAs have been used for many years to specify the quality of service delivered to corporate customers by for example telecom operators, as well as for regulating outsourcing contracts in the IT domain. However, SLAs have until recently not received much interest from the public at large.

Service Level Agreements come in a variety of forms. Today, a typical SLA states the obligations of the provider, the circumstances under which it is valid, and the penalties if the SLA is broken. An SLA often has both business and technical parts, but here we will focus on the technical part. To verify that the provider delivers service in accordance with the agreement, an SLA usually contains Quality of Service (QoS) parameters. QoS refers to the (measurable) ability of a system to provide the network and computation services such that the customer's expectations are met.

QoS have traditionally covered the dependability and performance characteristics of a service. Service dependability is usually defined as a combination

of the service availability (the proportion of time a system delivers service in accordance with the requirements) and service reliability (the system's ability to provide uninterrupted service). Service performance is usually characterized by throughput (the number of bits of data transmitted or processed per second) and delay (the number of seconds used for transmission or processing) [2]. The term QoS usually does not include security, even though several initiatives have tried to extend the term in this respect [1]. Today many service providers offer QoS guarantees as a part of their SLAs, however the focus in most cases is on availability. For example, the SLA for Amazon's Elastic Compute Cloud is in principle limited to the following statement: "AWS will use commercially reasonable efforts to make Amazon EC2 available with an Annual Uptime Percentage (defined below) of at least 99.95% during the Service Year." From the customers' point of view, the lack of guarantees of other non-functional attributes is a major drawback; e.g., a service with very low performance will be perceived as being unavailable.

3 The Need for Security SLAs

Even though service availability and performance are critical issues, security is often stated as the main barrier against outsourcing critical data and applications to actors where no previous trust relations exist. To mitigate the security risks associated with service oriented architectures, and to increase the trust in the providers, existing security mechanisms and their effectiveness should be formalized in contracts. Since the SLA is used to explicitly state the obligations of the provider, the implemented security mechanisms, their effectiveness and the implications of possible mismanagements should be a part of the SLA. This concept is also known as Quality of Protection (QoP); the ability of a service provider to deliver service according to a set of specific security requirements. In the following we will call such a contract a "security SLA". A security SLA should (at least) include:

- A description of the service that is to be provided
- The security requirements that the provider will commit to
- The process of monitoring security, including what evidence to collect and present, how the evidence will be collected and who will be responsible for it.
- The process of reporting problems, threats or security related incidents. This also includes information on what person to contact when a problem occurs and the acceptable time for resolving the problem.
- Consequences for cases when the provider (or customer) breaks the terms stated in the SLA, in terms of service credits or financial compensation. The service provider may also want to include constraints on the customer behavior and escape clauses defining when statements in the agreement do not apply.
- Legal and regulatory issues, including references to existing legislations and directives that may affect the service as well as the terms under which the SLA will not be valid.

To return to our UC example (Figure 1), the UC provider intends to offer a service to end-users that can satisfy a set of security requirements, and in order to accomplish this, it must establish a security SLA with each of the subcontractors. For the messaging service, assume that we have the following security requirements [1]:

1. User profile information must be stored in an encrypted state
2. Only a hashed value of the user password will be stored.
3. A user profile will only require a valid email address and username; age, gender, name, picture and phone number will be optional fields.
4. Information exchanged among the participants must be kept confidential
5. All text messages must be digitally signed
6. Authentication shall be based on symmetric encryption using a trusted third party as authenticator
7. The endpoints of all connections must be mutually authenticated
8. Only one instance of an authenticated user can participate in a communication session
9. Only the service provider will have access to statistical information
10. Asymmetric communication must be stored in an encrypted state and not for more than 48 hours
11. All location data must be logged for a minimum of 48 hours and maximum of 168 hours.

The security SLA also states that problems shall be reported to the messaging provider's contact email address, and that any security breaches and other security notifications shall be reported from the messaging provider to the UC provider's email contact point. General security monitoring shall be performed by an Intrusion Detection System (IDS), with specific filters to verify that no clear text user information, passwords or messages are transmitted. The IDS will detect unauthorized attempts to extract statistical information, and also verify that only one instance of an authenticated user will participate in a session, and check that all messages are signed. Spot-check audits are required to verify that user profile information and passwords are not stored in clear text, and that the logging requirements are fulfilled. In addition to the security requirements, the messaging service is subject to the European Data Retention Directive, requiring logging of communication endpoint IDs for six months.

4 Managing the SLA

Today's SLAs are usually in textual format; however, in the near future we envision an SLA as a dynamic construct, which will need to be renegotiated as the context changes [1]. There are several possible events that may trigger a change in the security terms; e.g., the user's needs may change during the validity period of an SLA, but more importantly, the provider's ability to provide a given level of security may change, e.g. due to recently discovered flaws in specific components.

We envision an SLA lifecycle as illustrated in Figure 2, which consists of six phases: Providers first publish their SLA templates, and users then initiate negotiation based on these. Once the negotiation is successful, providers and users formally have to commit to the resulting SLA, and the provider effectuates the provisioning. While the service is running, monitoring should ensure that the SLA terms are met, and when the relationship between user and provider comes to an end, the SLA must be terminated. Feedback loops are used when it is necessary to return to the negotiation phase from any of the active phases; e.g., if either of the parties cannot commit to a negotiated SLA, if the provider is unable to provision the service (due to overbooking), or if monitoring reveals that SLA terms are broken. In practice, the contracting period may vary from very short periods like a minute (e.g. the temporary lease of a fiber optic communication channel) to months and years for more stable services (e.g. a backup service for corporate data).

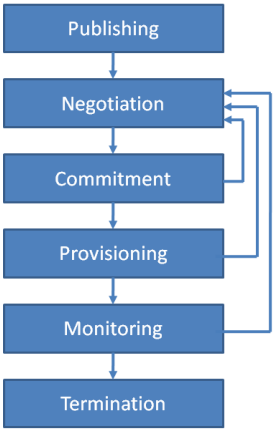

Fig. 2. The SLA lifecycle [1]

For composite services, security SLAs will need to be established between all actors that participate in the final service composition. The SLAs will therefore be negotiated and managed on several layers before a final service can be delivered to the customer; in fact, the customer need not even be aware that different service providers deliver these services.

Returning to the UC example, we assume a service delivery and security SLA have been established between the customer and the UC provider, providing voice, instant messaging, presence and conferencing services, with associated security service levels. For the UC provider, the risk of violating the security SLA towards the customer must be handled by ensuring satisfactory security SLAs towards its subcontractors. These agreements may be the results of a traditional Request for Quotation (RFQ) process, in which case the various service providers

have a static business relationship and security SLA with the UC provider, or it can be a more dynamic process where the UC provider queries potential service providers on demand. This is related to the publishing phase above.

The necessary negotiation phase is the same irrespective of the publishing scenario chosen. In this specific example, the UC provider has its own voice service, and as long as this can satisfy all voice-related requirements, the UC provider will not query or start negotiations with additional external providers. For the messaging service, provider B is the only one that fulfills all the security requirements listed, and is chosen over providers A and C. The same selection procedure is followed for the remaining services.

After the negotiation phase is completed, the UC provider initiates the commitment phase where the service providers commit to deliver the previously offered service and all parties digitally sign the security SLA. The necessary resources for service provisioning were reserved during the negotiation phase, and the service can easily be provisioned. The UC provider will then monitor the service fulfillment from the service providers, possibly using an external independent auditor, and breaches must be handled accordingly. This may result in renegotiation of the contract or eventually termination of the contract.

For the UC provider, establishing a security SLA towards its subcontractors minimizes the risk of violating the SLA they have committed to with their customers.

5 Technical Challenges

To facilitate the process of negotiating security SLAs with different potential service providers, to make the comparison of different service offerings simpler, and to simplify the commitment phase of the service lifecycle, there is a need for common industry standards and corresponding templates for machine-readable agreements [1]. However, there are no such templates for security SLAs available today.

Establishing a security SLA is not sufficient in itself; the agreed terms need to be monitored and controlled as well. However, monitoring and controlling security terms are inherently difficult. While other QoS aspects, such as the service availability, can easily be measured and controlled by the users themselves, security tends to be more difficult to monitor. One reason is the nature of service oriented architectures, which are designed to hide the inner workings of the services from the user, exposing only their APIs to the developers. Another reason is that the security requirements are often stated in terms of what should not happen, making it difficult to verify that the preventive mechanisms works as intended, until a breach has already occurred. In addition, the really clever attacks often go unnoticed.

6 Past, Present and Vision

The interests in and demand for security SLAs has varied throughout the years. Researchers began investigating security SLAs already in the beginning of the 90ies [3]. Early work on security agreements was performed by Henning [4], who already then raised the question whether security can be adequately expressed in an SLA. The interest in security SLAs received a new boost in the late 90ths when QoS agreements in IP-based network was a hot topic in the research community, especially in the telecommunication sector [5]. Researchers pointed out the need for security as a QoS attribute but did not manage to define service levels that were useful to users and service providers [6]. A second wave of interest was raised when the increased adoption of Service Oriented Architectures (SOA) drew attention to non-functional attributes, such as security and performance of web services [7–10]. More recently, the advent of the Cloud concept, which is characterized by elastic and on-demand measurable services, has given rise to a new demand for such agreements [1, 11–13].

To facilitate dynamic and automatic SLA negotiation between the service consumers and service customers, including renegotiation of SLAs, a machine-readable contract language will be necessary. The WS-Agreement specification [14] is a protocol for establishing SLAs between two parties, such as service providers and consumers. It is an open standard and it has been widely adopted for QoS support for service oriented architectures in web and grid contexts. WS-Agreement allows the use of any service term, and is therefore suitable for security agreements as well. The specification provides a template for the agreement, which consists of the name of the agreement (this is optional), the context (the participants and the lifetime of the agreement) and the agreement terms. The agreement terms are used to specify the obligations of the parties and the associated guarantee terms are used to provide assurance to the service consumer on the service quality and/or resource availability offered by the service provider. The current version of WS-Agreement does not include any ontology for incorporating security requirements in the SLAs, but the specification can relatively easily be extended with this feature [13].

7 Conclusion

We believe that the time of security SLAs has finally arrived. A security SLAs will not only ensure that the service consumer receives the required level of security, but may also put restrictions on the consumer, for example in terms of acceptable usage of the outsourced service. Effective security SLAs will also make sure that the service providers and consumers have a common understanding of the service terms. To achieve an open federated ecosystem of independent actors where services can be easily purchased, replaced and terminated whenever necessary, we need to see widespread use of machine-readable security SLAs. This will increase the uptake of service oriented architectures in new domains and foster innovation amongst developers, service providers, service customers, as well as end users.

References

1. Bernsmed, K., Jaatun, M.G., Meland, P.H., Undheim, A.: Security SLAs for Federated Cloud Services. In: Proceedings of the Sixth International Conference on Availability, Reliability and Security (AReS 2011). (2011)
2. International Telecommunication Union: Terms and Definitions Related to Quality of Service and Network Performance Including Dependability, ITUT E.800 (2008)
3. Irvine, C.: Quality of security service. In: Proc. ACM New Security Paradigms Workshop. (2000) 91–99
4. Henning, R.R.: Security service level agreements: quantifiable security for the enterprise? In: Proceedings of the 1999 workshop on New security paradigms. NSPW '99, New York, NY, USA, ACM (2000) 54–60
5. Grgic, I., Røhne, M.: Agreements in IP-based Networks. Telektronikk **2**(3) (2001) 186–212
6. Lindskog, S., Jonsson, E.: Adding Security to Quality of Service Architectures. In: Proceedings of the SS-GRR Conference. (2002) http://www.cs.kau.se/ stefan/publications/SSGRR02s/paper.pdf.
7. SLA@SOI Consortium: SLA@SOI (2011) `http://sla-at-soi.eu/`.
8. Righi, R.R., Kreutz, D.L., Westphall, C.B.: Sec-mon: An architecture for monitoring and controlling security service level agreements. In: XI Workshop on Managing and Operating Networks and Services. (2006)
9. Casola, V., Mazzeo, A., Mazzocca, N., Rak, M.: A SLA evaluation methodology in Service Oriented Architectures. In Gollmann, D., Massacci, F., Yautsiukhin, A., eds.: Quality of Protection. Volume 23 of Advances in Information Security. Springer US (2006) 119–130
10. Frankova, G., Yautsiukhin, A.: Service and protection level agreements for business processes. In: Young Researchers Workshop on Service. (2007)
11. de Chaves, S.A., Westphall, C.B., Lamin, F.R.: SLA Perspective in Security Management for Cloud Computing. In: Proceeding of the 2010 Sixth International Conference on Networking and Services, IEEE (March 2010) 212–217
12. mOSAIC Consortium: mOSAIC (Open source API and platform for multiple clouds) (2011) `http://www.mosaic-cloud.eu/`.
13. Meland, P.H., Bernsmed, K., Gilje Jaatun, M., Undheim, A., Castejon, H.: Expressing Cloud Security Requirements in Deontic Contract Languages. In: Proceedings of the 2nd International Conference on Cloud Computing and Services Science, (CLOSER). (2012)
14. Open Grid Forum: Web Services Agreement Specification (WS-Agreement) (2007)