Building the Security Foundation to Embrace Public Software-as-a-Service (SaaS)

Security Policies for SaaS Data Protection

Yuyu Chou[1], Jan Oetting[2] and Olga Levina[1]
[1]*Systems Analysis and IT, Berlin Institute of Technology, Berlin, Germany*
[2]*Consileon Business Consultancy GmbH, Karlsruhe, Germany*

Keywords: Software as a Service, Cloud Computing, Security Policy, Data Protection, Security Management.

Abstract: To mitigate the risk of confidentiality breaches when adapting public SaaS solutions, enterprises should build their security policies by setting up a system of security awareness. This paper presents a systematic approach to developing security policies, which includes the method and process used during the public SaaS system development life cycle. Hence, all employees will have the well-grounded concept to protect confidential data in the cloud.

1 INTRODUCTION

The security policy defines targets to be reached and serves as management's tool to meet business objectives in a public SaaS system as well as in in-house systems. By setting the rules for expected behaviour, security policies can protect information outside of internal IT system, and help a frim to comply with regulations as well as on-coming cloud legal requirements.

Information security policies may vary widely from company to company and no standard sample that can be copied (Diver, 2007). Firm size, culture, industry sector, country, regulation, the business process and the IT support will influence the statements. Among those companies which have security policies are often driven by compliance, not from their own convictions; therefore many companies do not have a fit security policy, and their security policies only used to a small extent (Oracle, 2009).

Up to now, most of the companies deploying Cloud Computing solutions have no cloud-specific security policies and procedures in place (Hickey, 2010). It is essential for enterprises to be equipped with reliable security policies, to have a systematic and an objective judgement for managing a SaaS system development life cycle (SDLC). There is already some literature suggesting guidelines for creating general security policies, e.g., from (ISC)[2] (Tipton and Henry, 2007), SANS Institute (Diver, 2006), and addressing the importance of Cloud security policy (Jaeger et al., 2008); (Hickey, 2010). However, there is little literature guiding enterprises how to build security policies to reduce the risk of using public SaaS services. To solve the problem, this paper presents a method and process for companies to develop cloud specific security policies during an SaaS SDLC.

The structure of the remaining part of this paper is as follows: Section Two presents the concept of a security policy set; it enables enterprises to build different security policies for different users and purposes. In Section Three, the policy development and management is introduced. The approach of applying the policy set based on the SaaS SDLC is suggested in Four Section. Finally, conclusion and work-in-process are presented in Section Five.

2 POLICY TYPE FOR SAAS SDLC

For an enterprise to develop a single good policy document that addresses all types of users and information security controls is very complicated (Diver, 2007). Different users will need different documents. Therefore, for building effective and systematic policies to embrace a new SaaS integrated IT system, this paper suggests using the

Chou Y., Oetting J. and Levina O..
Building the Security Foundation to Embrace Public Software-as-a-Service (SaaS) - Security Policies for SaaS Data Protection.
DOI: 10.5220/0004024702270232
In *Proceedings of the International Conference on Security and Cryptography* (SECRYPT-2012), pages 227-232
ISBN: 978-989-8565-24-2

concept of a hierarchical security policy set (Diver, 2006). It can help enterprises build solid and clear security policies step by step under tight resource conditions, rather than compiling unmethodical and scattered security rules. Table 1 summarizes the characteristics of these three level policies.

Table 1: Different types of security polices.

	Address	Audiences	Update frequency
Governing policy	What	Managers, technical members, end users	Low
Technical policy	What, where, who, when	Technical members	Middle
Detailed requirement	What, where, who, when, how	End users, developers, service providers	High

2.1 Governing Policy (Level One)

The security governing policy is on top of the security policy framework. Security governing policy should focus on desired results, not on means of implementation (Guel, 2007).It is designed to set out clearly the strategic aims and control objectives that will guide the security development of the new cloud integrated IT systems. Organizations should aim for writing their security policy on not more than three pages of A4 (Calder et al., 2010). Detailed policy statements should be left out and be written on the second or third level of the documentation. In this way the governing policy can remain constant over a longer period of time without conflicting with the regular technology changes.

2.2 Technical Policy (Level Two)

Technical policy includes system-specific and functional policies. It should cover a part of the governing policy topics, but it is more precise than the governing policy. The statements should depend upon the business needs, and focus on the different domains of security control belonging to general technical topics, e.g., access control, applications development policy. Detail requirements should be defined in level three policies because these level two policies will be applied by technical custodians as they carry out their system security responsibilities, or they will be used for specific applications. Following, a number of SaaS topics to be addressed in the technical policy are given:

First, when defining requirements for external cloud services, cloud specific requirements for providers (BSI, 2010), legal requirements, and company specific rules for critical data protection should be observed. Companies have to be careful with too detailed statements on this technology-independent level, because valid options could be ruled out without noticing it.

Second, extra internal compensating security controls should be implemented if needed. IT and security teams should do the corresponding tests and implementations when selecting an SaaS solution. Data should be protected from the source or before sending out to the servers of a public SaaS provider.

Third, set up mobile usage policy for using cloud web application. Mobile applications have different risks compared to a traditional web application model (OWASP, 2011). The policy should therefore additionally define what kind of business processes may be accessed by mobile devices and what kind of data may be processed and stored on them.

2.3 Detailed Requirement (Level Three)

Level three policies should give detailed guideline directions for employees "how" to carry out the policy statements (Diver, 2007). Based on the survey of the Computer Security Institute, roughly one in four believed that more than 40 percent of their total financial losses from attacks were due to insider activities (CSI, 2007). Implementing precise and comprehensive policies is the solution for dealing with insider threats like decision errors, skill-based errors, perceptual errors and routine violations (Predd et al., 2008). Therefore, both external and internal threats could be greatly alleviated when enterprises know how to build and execute level three policies well. Furthermore, these documents can act as a backup facility; once a staff leaves, the knowledge will not be lost and the policy requirements can still be carried out.

The relationship between level two and level three policies can be none, one-to-one or one-to-many (Diver, 2006). The following explains the types of level three policies:

- **Guideline.** Guideline is not a required element of a policy framework, but it helps users to understand the complete scope of flows and options. If an item is considered mandatory, then this should be defined in higher level policies, standards or baselines.
- **Standard.** Standards provide the specification of the technology to effectively enable the organization to comply with the policies and provide interoperability within the enterprise through the use of common protocols (Tipton and Henry, 2007). It

also can be referred in many existing public standards.

- **Procedure.** Procedures are step by step instructions (Guel, 2007). They can reduce cloud risks by indicating how the policy should be implemented and who should do what in order to accomplish the tasks as an effective and consistent method.
- **Baseline.** It provides a description of how to implement and stipulate the minimal requirements to secure the data internally as well as in clouds. The requirements should be consistent throughout the organization. Usually, the baselines are specific rules supporting the policy and standards that have already been developed.

3 MANAGEMENT OF A SECURITY POLICY

Policy Management is the process of managing and maintaining policies effectively within the organization (Rasmussen, 2011). Developing good security policies for managing SaaS systems, the developers need to consider business, legal, technical, cultural and security aspects.

3.1 Policy Development and Management Process

To avoid being "blind men feeling the elephant", this paper proposes nine steps of policy development process. This management cycle is started with the "preparation" and ends with an integrated "periodic review and update policy" into the security management process.

3.1.1 Step One: Preparation

There are some preparations that should be taken before the kick-off of the security policy development project. First, understand the business goals and direction. Second: determine resource involvement. Designing the governing security policies is usually more complicated than others. Therefore, the primary involvement for a governing policy might be just the security team, and the optimum size for the core team is around 3 people (Kee, 2001). The members can be external employees or security consultants, but it needs to be made sure that they can be trusted and signed a non-disclosure agreement (NDA).

3.1.2 Step Two: The Project Kick-off and Top Management Buy-in

It may take longer time, if security policies are defined for the first time. Therefore, same with other important security projects, the top managers' acceptance and support for the policy development is the stepping stone. Without the commitment of top management, the project will be given up halfway or will easily be a perfunctory policy.

3.1.3 Step Three: Reviewing of Existing Policy

Many enterprises may not have a security policy. But if a company has existing security policies, even though they are weak, a review will help when developing a new security policy.

3.1.4 Step Four: Data Gathering

The policy development team needs to know who the potential attackers are, what they can do against them and decide how to enhance security to protect the information assets in internal systems as well as in the cloud. A number of security standards, guidelines and cloud articles can offer the team sufficient cloud security knowledge. Besides, clearly understanding cloud related industrial standards, national or local legal requirement, the company resource, risk appetite from managers, technical limitations, and SLA as well as contract from SaaS providers is also important. Hence, the policy development team can gather sufficient information to define rational and enforceable policies which links to internal and external mandates for staffs to comply.

3.1.5 Step Five: Writing Initial Draft

Six suggestions are provided below:

First, mandatory: Policies are expected to be complied with; therefore statements including words such as "must", "will", "need to" are better than using weaker directives such as "should", "may" or "can" (Tipton and Henry, 2007).

Second, keep to a minimum length. The length of each policy should be limited (Calder et al., 2010); otherwise few employees will read it attentively.

Third, do not use abstract language. A quantitative description is usually better than qualitative one.

Fourth, ensure that roles and responsibilities are defined and clearly understood. Easy to understand instructions are very important because the aim of

security policy is to communicate with both technical and non-technical users (in technical requirements it is acceptable to use technical terms).

Fifth, rules must be followed and enforceable; otherwise, employees will try to work around them.

Sixth, when formulating the policies, it should be mentioned if failure to comply with the policy will result in disciplinary action and punishment; otherwise, the policy will not be thoroughly executed.

3.1.6 Step Six: Reviewing Process

The completed draft security policy must be reviewed, and feedback must be asked for from the appropriate parties such as executive, business leaders, IT and security managers, human resources, legal staff, audit departments and employee union. Make sure the policy is closely aligned with existing as well as future Human Resource guidelines and other company policies. In addition, make sure the policy is understandable for end users. Typos or errors that are not seen by the authors should be corrected. Last but not least, before publishing policies, the security team should check if any policy statements are not currently in force or only partially exist in the enterprises organization (Diver, 2007). To document these gaps and to determine which groups or individuals are responsible for closing them, can make the implementation more efficient.

3.1.7 Step Seven: Final Sign-on and Publish

The final review and sign-off is typically accepted by all management levels, and done by the business leaders, the chief security officer (CSO) or IT Managers. However, publishing and communicating with employees can be a challenge, in particular when the policy changes. Hence, there are four suggestions shown below, which may be helpful to efficiently publish the security policies and communicate with employees.

First, make it as a contractual requirement.

Second, security awareness should be part of required training. Campaigns must convey the seriousness of new cloud related specific policies, and explain the reasons why it is necessary. Otherwise, employees will not change their behaviour, even if the new process is more beneficial and easier.

Third, Email is a good way to inform employees about security policy changes quickly.

Fourth, documents should be easily accessible and available for download, printing as well as saving. However, if the content contains confidential information, these documents should be only accessible by the security team and properly secured from general distribution (Tipton, 2007).

3.1.8 Step Eight: Compliance and Enforcement

Education and training are the best way to make employees understand how to comply with the policies. Once the compliance grace period is exceeded, and the policy is enforced, auditors should ensure that policies are surely followed. Management should take care that based on one of the principle “separation of duties”, the auditors monitoring compliance with the security policy should not be the persons who implemented the policy. Policies should be enforced strictly and noncompliance should be punished (Kee, 2001).

3.1.9 Step Nine: Regularly Review and Update

Security policy needs to be updated when for example, compliance with new cloud regulatory requirements, new important project rollouts, customer request, technology change, and editorial errors occur. To ensure policies do not become out of date, the security team should review the high level policy at least every 2-3 years, and detailed requirements every year (Guel, 2007). In case an SaaS solution is used and security policies are established, the security team has to ensure that the policies related to such as data handling and access control are addressing the Cloud specific logic.

3.2 Good Security Policy Requirements

Characteristics of good data protection policies can be hierarchical (Petrocelli, 2005) and should reach the following requirements:

- They must be written down (Petrocelli, 2005).
- They must be specific to the organization, but not be cloud vendor specific.
- Roles and responsibilities must be defined in accordance with the security policies (ISO/IEC, 2005).
- They must be easy to understand.
- They must achieve a balance between current practice and preferred future (Diver, 2007).
- They must develop sanctions for non-compliance. (Tipton and Henry, 2007).
- They must provide sufficient protections against

all threats, and the statements should also be implementable, measureable and enforceable.

4 APPLIED IN AN SAAS SDLC

Managers often know the importance of security, but run it with a low priority (Martin, 2011). Although public SaaS providers usually offer non-negotiated service level agreements (SLAs) and generally they will not customize their service to fulfill the security requirements of each client (ENISA, 2009), the security level of the client company should not be downgraded because of adapting a new public SaaS application.

An SaaS system development life cycle (SDLC) is composed of nine phases (Chou et al., 2011), as shown in Figure 1. From the beginning of the SaaS system design preparation, managers should integrate the security concepts based on defined security policies into development, thus the system can be securer and more cost-saving (BSI, 2008) than ameliorate it afterwards. The following explains briefly how a company uses these policies to build a data secured SaaS SDLC.

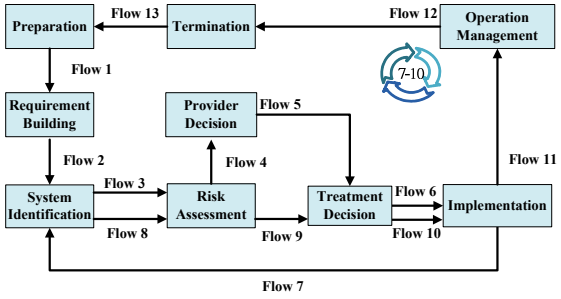

Figure 1: SaaS system development cycle (revised from Chou et al. (2011)).

First part, during the phases "Preparation" and "Requirement Building". When a company wants to adapt a first SaaS solution, either the selected SaaS solution can follow the existing governing policy, or the governing policy should be updated. It is important to build the security policy set before using an SaaS system, although this is hard to fulfil for many of the companies. Enterprises should at least set down or update the governing policy before starting the SaaS system development cycle. Other policies can be verbally agreed on by managers, and should preferably be written down. All candidates need to be checked if they can reach the vendor requirement baseline, and can follow the IT service outsourcing policy. The supportive documents for data protection in the SaaS system development life cycle are shown in Table 2.

Second part, during the phases "System Identification" and "Risk Assessment". These phases establish the base for the decision in field of business, technique, security, and legal aspects. The responsibility of the security team is: to check if related internal systems miss any control based on security requirements and test the candidates' solutions against standard security weaknesses. In addition, the security team should clarify the security responsibility among the internal team and SaaS providers. Then co-work with the IT team to execute the basic risk evaluation based on the risk assessment procedure and report the risk summary from each solution to managers.

Table 2: Suggested supportive security polices during SaaS system development life cycle.

Phase	Supportive policy documents
Preparation	Security governing policy, vendor requirement baseline, applications development policy
Requirement building	Data classification guidelines, data protection policy
System identification	Web security testing procedures, security control baselines, IT service outsourcing policy
Risk assessment	Risk assessment procedure (method)
Solution decision	Outsourcing service approval process
Treatment decision	Cryptography policy, network security management policy, access control policy, etc.
Implementation	System testing procedure
Operation management	Operational procedures, disaster Recovery policy, equipment usage policy, audit policy, change control procedures, business continuity management policy, etc.
Termination	System termination guideline or procedure

Third part, during the phases of "Treatment Decision" and "Implementation". Once the managers decide to use an SaaS solution, this SaaS solution needs to be further checked in detail if the solution has any vulnerabilities, can comply with the data protection policy, and if it follows IT service outsourcing policy. Once unacceptable risks in a decided SaaS application are found, extra compensating security controls should be designed during the treatment decision. Thus, these extra controls can provide an equivalent or comparable level of protection for the system when data processed, stored or transmitted (NIST, 2009). After the implementation, the security team needs to check if the new SaaS integrated system is under risk acceptable status based on security testing procedure. If the result is not acceptable, the development team needs to go back to "Identification Phase" and run the cycle (7-10

Flows) again to decide new security treatments.

Fourth part, during the ”Operation Management” and “Termination” phases. After the successful completion of the system go-live test, operational procedures should be delivered to end users. Then, this system is going to the normal operation status of security management process. Before the application phases out, employees should have the termination guideline or procedure on hand. Therefore, the data can completely and without harm be moved back to the internal system or to other systems.

5 CONCLUSIONS

Although security policy is only one of the security controls, this is the fundamental base for building a secure public SaaS system development life cycle. To solve the problem of confidentiality breaches in public SaaS solutions, a company needs to have multiple layers of defense and strategies against potential threats. These strategies must be consistent with the business needs, be well defined in the security policy and be effectively published for every employee to comply. Based on the process and the methods shown in this paper, companies can proceed step by step and build their policy systematically even under tight resource conditions. Therefore, the customer data, employee data and confidential business information will be better protected during the whole SaaS system development life cycle. Our on-going work is to build an automatic data protection tool based on the enterprises' data classification policy. This control is independent from the support of public SaaS providers; thus, the tool can enforce the security policy requirements and help to avoid confidentiality breaches. To sum up, although there are many security control objects, having a good tailored security policy set is the first priority for enterprises before using SaaS applications. Based on the well-defined and executed security policies, companies will not only take advantage of using an SaaS solution, but will also protect their data on the business battlefield.

REFERENCES

BSI, 2008, *Information security management systems (ISMS)*, BSI-Standard 100-1, v 1.5. German Information Security Agency, Bonn.

BSI, 2010, *Minimum security requirements for Cloud Computing providers*, draft BSI Standard (in German), German Information Security Agency, Bonn.

Calder, A., Watkins S. and Watkins S.G., 2010, *Information security risk management for ISO27001 /ISO27002*, IT Governance Ltd, UK.

Chou, Y., Levina, O., and Oetting, J., 2011, 'Enforcing confidentiality in a SaaS cloud environment' *Proceedings of the 2011 19th Telecommunications Forum (TELFOR)*, 22-24 Nov. 2011, pp. 90-93, IEEE Digital Library, IEEE Portal.

CSI, 2007, Computer crime and security survey 2007, Computer Security Institute, viewed 20 Oct. 2011, < http://www. gocsi.com/forms/csi_ sur vey.jhtml >

Diver S., 2007, Information security policy –a development guide for large and small companies, SANS Institute, viewed 28 February 2012, < http://www.sans.org/reading_room/whitepapers/policyissues/information-security-policy-development-guide-large-small-companies_1331 >

ENISA (European Network and Information Security Agency), 2009, Cloud Computing: benefits, risks and recommendations for information security.

Guel, M. D., 2007, A short primer for developing security policies, SANS Institute.

Hickey, A. R., 2010, Cloud computing security policies, procedures lacking, CRN, viewed 28 February 2012, <http://www.crn.com/news/security/224201359/cloud-computing-security-policies-procedures-lacking.htm>

ISO/IEC 2005, *Information technology – security techniques – information security management systems – requirements*, ISO/ IEC 27001 Standard.

Jaeger, P., Lin J., and Grimes, J., 2008, 'Cloud Computing and information policy: computing in a policy Cloud?', *Journal of Information Technology Politics* , vol. 5, no.3, pp. 269-283.

Kee, C. K., 2001, Security policy roadmap - process for creating security policies, NANS Institute.

Martin, E., 2011, 'What's wrong with security?' *Information Security*, vol.13, no. 9, November, pp.8-10.

NIST, 2009, *Recommended security controls for federal information systems and organizations*, NIST SP 800-53 v3 Standard.

Oracle, 2009, Securing data at the source: a guide to oracle database security, viewed 28 February 2012, < http://media.techtarget.com/Syndication/SECURITY/SecuringDataSource.pdf>

OWASP, 2011, Mobile top 10 risk, viewed 28 February 2012,<https://www.owasp.org/index.php/OWASP_Mobile_Security_Project#tab=Top_Ten_Mobile_Risks >

Predd, J., Pfleeger, S. L., Hunker, J. and Bulford, C., 2008. 'Insiders behaving badly', *Journal of IEEE Security & Privacy*, vol. 6, no. 4, July/August, pp. 66 – 70.

Rasmussen, M., 2011, Accountability in Policy Management, Corporate Integrity, viewed 28 February 2012, < http://www.corp-integrity.com/compliance-management/accountability-in-policy-management>

Tipton, H. F. and Henry, K., 2007. *Official (ISC)^2 Guide to the CISSP CBK*, Auerbach Publications, New York.