Seeding the Cloud: An Innovative Approach to Grow Trust in Cloud Based Infrastructures

Stéphane Betgé-Brezetz[1], Aline Bousquet[3], Jérémy Briffaut[3], Eddy Caron[2],
Laurent Clevy[1], Marie-Pascale Dupont[1], Guy-Bertrand Kamga[1], Jean-Marc Lambert[4],
Arnaud Lefray[2,3], Bertrand Marquet[1], Jonathan Rouzaud-Cornabas[2],
Lamiel Toch[2], Christian Toinard[3], and Benjamin Venelle[1]

[1] Alcatel-Lucent Bell Labs, France
{stephane.betge-brezetz,laurent.clevy,marie-pascale.dupont,
guy-bertrand.kamga,bertrand.marquet,
benjamin.venelle}@alcatel-lucent.com
[2] University of Lyon, LIP Lab, UMR CNRS - ENS Lyon - INRIA - UCB Lyon 5668, France
{eddy.caron,arnaud.lefray,lamiel.toch,
jonathan.rouzaud-cornabas}@inria.fr
[3] ENSI Bourges, LIFO Laboratory, France
{aline.bousquet,jeremy.briffaut,
christian.toinard}@ensi-bourges.fr
[4] Gemalto, France
jean-marc.lambert@gemalto.com

Abstract. Complying with security and privacy requirements of appliances such as mobile handsets, personal computers, servers for customers, enterprises and governments is mandatory to prevent from theft of sensitive data and to preserve their integrity. Nowadays, with the rising of the Cloud Computing approach in business fields, security and privacy are even more critical. The aim of this article is then to propose a way to build a secure and trustable Cloud. The idea is to spread and embed Secure Elements (SE) on each level of the Cloud in order to make a wide trusted infrastructure which complies with access control and isolation policies. This article presents therefore this new approach of trusted Cloud infrastructure based on a Network of Secure Elements (NoSE), and it illustrates this approach through different use cases.

1 Introduction

Cloud Computing is revolutionizing the delivery of information services as it offers several advantages over traditional IT systems mainly in terms of cost-reduction and agility. However, it raises many concerns related to security and privacy which are main obstacles for its large adoption [4].

Moreover, the current situation of Cloud Computing is dominated by the model where the Cloud Service Provider (CSP) defines his terms/conditions and each Cloud User (CU) has to accept them and trust the CSP. Unfortunately, several recent cases of security or privacy breaches [5] showed that CUs should (i) pay attention to how CSPs effectively manage security and privacy within their infrastructures and (ii) be

A. Galis and A. Gavras (Eds.): FIA 2013, LNCS 7858, pp. 153–158, 2013.

part of this management. This point is underlined by key standard [6] or regulation bodies notably in Europe [7]. Indeed, under the EU laws, the CU remains responsible for the collection and processing of sensitive data, even when third parties (i.e., CSP) process these data. Although he has currently no regulatory responsibility, the CSP has however to comply with the contractual obligations required by the CU and therefore has to provide the CU with the necessary features and technologies allowing the protection of the CU sensitive data within his Cloud infrastructure.

Several works are then tackling these Cloud privacy and security issues. Regarding data protection, we can mention the Data Protection as a Service approach [1] or platforms for Cloud privacy management [2]. However, these approaches do not really allow defining a large range of privacy policies and the data privacy protection mechanism needs also to be securely executed so that it should not be corrupted. Regarding the protection of software execution, the use of Secure Elements (SE) is a common solution. A SE can be for example a Trusted Platform Module (TPM) with a secure crypto-processor providing secure storage for cryptographic keys and which can also be used to verify the validity of the software running on a computer [12]. Similarly, Hardware Security Module (HSM), a special type of secure crypto-processor has been used to manage digital keys, and to provide strong authentication to access critical keys for server applications [13]. Several research projects [14] have been launched to define new ways to grow trust in the cloud but not by directly embedding SE (e.g., A4Cloud, TClouds) or using SE (e.g., TECOM) but not tackling how they can build together a cloud Trusted Computing Base.

The European CELTIC+/Seed4C project (Secure Embedded Element and Data privacy for Cloud) [16], presented in this paper, aims to address these key challenges of Cloud security and privacy management. In this Seed4C project, we propose a new approach of cooperative Secure Elements (or “Seeds”) allowing the trusted enforcement of a wide range of policy in order to ensure the security properties (confidentiality, integrity, availability) and to protect the privacy of user sensitive data. In order to achieve this, Seed4C aims to tackle various questions and technical challenges such as: What are the critical functions to be supported by these SEs? How to distribute SEs in the infrastructure and provide added value to platform and services? How to address communication between SEs and from SEs to embedding machines? The project is then proposing a technology able to address these challenges and that will be further carried out in use-cases of different domains.

This paper is then structured as follows. In Section 2, we explain the key principles of the Seed4C approach and how it can grow trust for Cloud infrastructures. In Section 3, we illustrate this approach through two examples of use cases respectively in the domains of high performance computing and privacy policy enforcement. Finally, in the conclusion, we discuss the next stages of our work.

2 Seed4C Approach: The NoSE as a Minimal Trusted Computing Base for the Cloud

The Network of Secure Elements (NoSE) is the cornerstone of the Seed4C project. NoSE is then a network of interconnected Secure Elements defining, for the Cloud, a minimal Trusted Computing Base (TCB) (i.e., a minimum set of hardware, firmware, and/or software components necessary for security) [11].

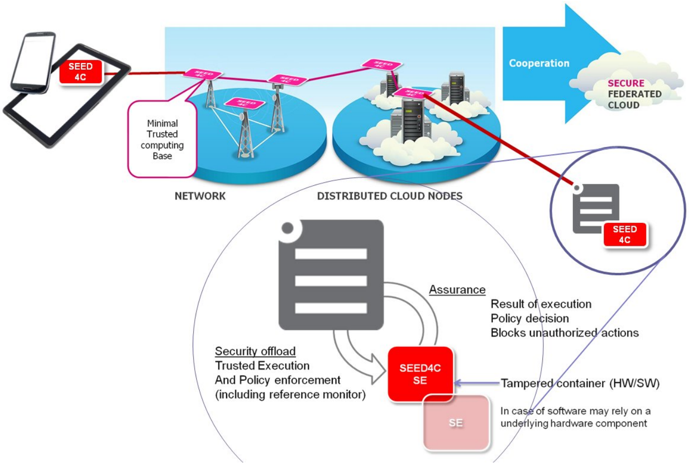

Fig. 1. Seed4C approach based on a Network of Secure Elements ("seeds") deployed within the Cloud infrastructure and offering trustable security and data protection

Secure Elements (SEs) are software or hardware elements that are usually connected locally with/in pieces of equipment. They do not communicate apart with some management services through a dedicated and controlled channel on the overall system or infrastructure. As said, SEs are by definition and construction elements that do not communicate or in a very controlled way. Making them communicate and collaborate to build a Network of Secure Elements that can secure an entire Cloud based infrastructure is indeed a research challenge.

In the Seed4C project, we will introduce deployment and configuration ability at the architecture level. With the NoSE, each Secure Element represents locally the minimal TCB to implement security function and the interconnection of SE makes the minimal TCB for a trusted service on a Cloud infrastructure. The trusted Cloud Computing Base allows connecting different parts of the infrastructure from the access and transport network until the Cloud nodes.

The NoSE allows end-to-end virtual security chain from the user terminal to the high-end server where the services are executed or to the place where data are processed and stored (see Figure 1). One important aspect of the NoSE is proximity. For this purpose, different options of embedment of Secure Element within the Cloud infrastructure will be considered (e.g., SE embedded in a physical Cloud node or rack, SE connected to a node hypervisor, or even software SE embedded in a Virtual Machine). Moreover, as a NoSE is composed of several elements, we can easily envisage that the NoSE will scale accordingly to the Cloud infrastructure or network. Its proximity to the running Virtual Machine or virtualized execution environment will allow correctly checking the environment and giving high assurance. The SE as element of the NoSE will be managed during the equipment lifecycle, for the physical provisioning part, but will be loaded/unloaded of some critical assets (credentials, attributes, etc) through the NoSE protocols. The SE will be accessed by local middleware/services during the life of a service from designing, provisioning & decommissioning of their components. The NoSE independence will provide a separate channel of trust. The policies / security rules will be executed by the different components of the architecture and the SE will contribute to that execution. SEs will communicate inside the NoSE using a secure channel such as the one specified by the ETSI Technical Committee Smart Card Platform [15].

As a first result of the project we will rely on a security language [9] in order to provide a canvas of easy to instantiate security properties. The application developers will be able to use these high level security properties. Through a specific compiler, the security properties will be translated into security policies for the different protection mechanisms at each level of the Cloud. These policies will be transferred through the NoSE to each local enforcement point. The policies will be then enforced by a set of reference monitors such as SE Linux and PIGA [9]. These reference monitors, embedded in SE and linked each other, will collaborate through the NoSE. Evidences of enforcement will finally be sent back to the CU thanks to the NoSE.

3 Examples of Addressed Use-Cases

3.1 High Performance Computing

When one spokes about High Performance Computing (HPC) one thinks about clusters and grids. Nevertheless since few years Cloud Computing offers services for scientists who need high performance computing facilities for their researches. Such intensive computing can use very sensitive data notably in domains as eHealth.

For ensuring the adequate data projection on the computing infrastructure, the software (middleware) which is in charge of communications between scientists and services is then critical. We will use the SOA based middleware DIET developed at ENS Lyon [8]. The principles of DIET are simple and the programming model is based on the GridRPC paradigm. A user sends a request to DIET to know whether a service is available on the platform. If this service exists, he sends the data needed by the service without carrying about where it is located, and finally obtains the results.

Recent developments on the DIET middleware stretch to make it compatible with some CSPs like EC2, Eucalyptus, Nimbus, OpenStack, OpenNebula, etc. Furthermore, DIET was not originally designed to address security problems as confidentiality and integrity. We propose therefore a proof of concept to add security skill to this middleware by leveraging the SE capabilities so that the computation on sensitive data will only be assigned to a trusted node having the adequate security protection. For instance a DIET user in an aerospace company wants to compute a model of an airplane and he does not want the rival companies to know what he does. So his requests and data sent must be confidential (confidentiality) and not allow unauthorized modifications (integrity). Thanks to the SEs embedded both in 1) the servers' daemons which provide services and 2) virtual machines instantiated by these servers, security properties are guaranteed.

3.2 Privacy Policy Enforcement

Another domain that will benefit from the NoSE approach of Seed4C is privacy. Indeed, in the Cloud context, as the CU outsources sensitive data (end-users personal information or enterprise documents such as HR documents on employees, bids, tenders, patents, etc.) to the CSP, the CU should be able to specify some requirements related to access control, data location, retention management, data usage/motion tracking, etc., governing how his data and the related processing must be protected within the Cloud infrastructure of the CSP. Moreover, in order to strengthen the trust between both parties, the CSP must be able not only to enforce the CU data protection requirements all along the data lifecycle but also to prove this enforcement.

For this purpose, we have proposed an approach of multilevel privacy policy enforcement [3] which consists for the CU to express his privacy and data protection requirements as policies applicable at several levels. One can distinguish for instance the *Application-Level* that includes policies governing the end-user or application level actions on CU data and the *Infrastructure-Level* that includes policies governing the CSP components (e.g. storage system, file system) actions on CU data.

In this multilevel policy context, the NoSE concept will be a real chain of trust between the CU and the CSP. Each Policy Enforcement Point either controlled by the CU (*Application-Level Policy Enforcement*) or by the CSP (*Infrastructure-Level Policy Enforcement*) could be executed within or in cooperation with a Secure Element that can among other provide certified information as needed, evaluate policies and generate the tamper-proof traces that can help to prove the fulfillment of the CU data protection requirements.

4 Conclusion

The distributed and dynamic nature of Cloud infrastructures requires thinking differently on how security and assurance are brought to execution elements. The Seed4C project, presented in this article, proposes a new approach called the NoSE (Network of Secure Element). The NoSE can be beneficial in several ways, bringing trust from the infrastructure layer up to the service and application ones.

At this stage, several use-cases have been identified and it is being elaborated the architecture and the abstract model to express policies. The next step is to define how to distribute policies to the NoSE and how the NoSE exchanges information, validates policies, indicates compliance and provides assurance. Moreover, the project has also planned to validate the Seed4C platform using real and large scale distributed infrastructures such as Grid'5000 [10] which can be used to deploy our own Cloud infrastructure with a NoSE. Also different exploitation models will be studied, from the cloud infrastructure operators that control the NoSE up to Over-The-Top scenarios that benefit from the NoSE at lower layers.

Acknowledgements. The authors would like to thank all the Seed4C partners for their valuable contributions. See [16] for more on Seed4C and its consortium.

References

1. Song, D., Shi, E., Fischer, I., Shankar, U.: Cloud Data Protection for the Masses. IEEE Computer Magazine 45(1) (2012)
2. Pearson, S., Shen, Y., Mowbray, M.: A Privacy Manager for Cloud Computing. In: Jaatun, M.G., Zhao, G., Rong, C. (eds.) CloudCom 2009. LNCS, vol. 5931, pp. 90–106. Springer, Heidelberg (2009)
3. Betgé-Brezetz, S., Kamga, G.B., Dupont, M.P., Ghorbel, M.: Privacy Control in the Cloud based on Multilevel Policy Enforcement. In: IEEE 1st International Conference on Cloud Networking (CloudNet 2012), Paris, November 28-29 (2012)
4. Srinivasamurthy, S., Liu, D.Q.: Survey on Cloud Computing Security. In: Proc. Conf. on Cloud Computing, CloudCom 2010 (2010)
5. Rashid, F.Y.: Epsilon Data Breach Highlights Cloud Computing Security Concerns, http://eWeek.com (2011)
6. Jansen, W., Grance, T.: Guidelines on Security and Privacy in Public Cloud Computing. NIST (2011)
7. Article 29 Data Protection Working Party, "Opinion 05/2012 on Cloud Computing", WP 196, Brussels (July 2012)
8. Caron, E., Desprez, F.: DIET: A scalable toolbox to build network enabled servers on the grid. International Journal of High Performance Computing Applications 20(3) (2006)
9. Afoulki, Z., Bousquet, A., Briffaut, J., Rouzaud-Cornabas, J., Toinard, C.: MAC protection of the OpenNebula Cloud environment. In: International Conference on High Performance Computing and Simulation (HPCS), July 2-6 (2012)
10. Bolze, R., et al.: Grid'5000: A large scale and highly reconfigurable experimental grid testbed. International Journal of High Performance Computing Applications 20(4) (2006)
11. NCSC DoD/NIST Orange book Part I section 6.3 (December 1987), http://www.kernel.org/pub/linux/libs/security/Orange-Linux/refs/Orange/OrangeI-II.html#toc6
12. http://www.trustedcomputinggroup.org/resources/trusted_platform_module_tpm_summary
13. http://www.opendnssec.org/wp-content/uploads/2011/01/A-Review-of-Hardware-Security-Modules-Fall-2010.pdf
14. http://cordis.europa.eu/fp7/ict/security/projects_en.html#CLO
15. http://www.etsi.org/deliver/etsi_ts/102400_102499/102484/07.00.00_60/ts_102484v070000p.pdf
16. http://www.celtic-initiative.org/Projects/Celtic-Plus-Projects/2011/SEED4C/seed4c-default.asp