An Approach for Securing Critical Applications in Untrusted Clouds

Luigi Coppolino*, Salvatore D'Antonio*, Giovanni Mazzeo*, Gaetano Papale*, Luigi Sgaglione* and Ferdinando Campanile†
*University of Naples "Parthenope"
Department of Engineering, Naples, Italy
{luigi.coppolino, salvatore.dantonio, giovanni.mazzeo, gaetano.papale, luigi.sgaglione}@uniparthenope.it
†Synclab s.r.l
f.campanile@synclab.it

***Abstract*—The cloud computing has recently emerged as compelling paradigm for managing and delivery services over the internet. However, users as well as critical infrastructure operators, have legitimate concerns about the confidentiality, integrity and availability, in short the dependability, of applications and their data hosted on a third-party cloud. The dependability is become a commercial imperative for cloud providers, especially to support cloud computing for critical infrastructures. In this paper the SecureCloud project, its approach and goals are presented. SecureCloud aims to remove technical impediments to dependable cloud computing, encouraging and enabling a greater uptake of cost-effective, environment-friendly, and innovative cloud solutions, in particular, for critical infrastructure applications.**

***Keywords*-Trusted Cloud Computing; Intel Software Guard eXtension (SGX); Critical infrastructures.**

I. Introduction

Confidentiality, integrity, and availability of applications and their data are requirements that immediately concern to almost all organizations that use cloud computing. This is particularly true for organizations that must comply with strict confidentiality, availability and integrity policies, including those society's most Critical Infrastructures, such as finance, utilities, health care and smart grids. Dependability (which implies confidentiality, integrity, availability) has emerged as a commercial imperative for cloud providers to be able to support emerging markets as well as cloud computing for critical infrastructures or cloud robotics. The cloud has not only become a critical infrastructure itself but it needs to support other critical infrastructures. These include smart grids and systems in the health and transportation domains but also extend to future large-scale computing, such as the Internet of Things (IoT) and Cyber-Physical Systems (CPS). The lack of sufficient dependability, however, is increasingly becoming the primary barrier to the broad adoption of cloud computing, not only in the critical infrastructure domain but also in all domains in which the survival of a company depends on the reliability of the cloud. Hence, the cloud becomes itself a critical infrastructure for which we need to guarantee adequate dependability such that we can justifiably place our trust in the hosted applications. The Cloud Industry Forum (CIF) [**?**] reports that data security and privacy - which are two important factors of dependability - are the top two concerns preventing the adoption of cloud computing in the United Kingdom. More in general, in Europe, the Steering Board of the European Cloud Partnership (ECP) identifies data protection and information security concerns as "the most ubiquitous requirements" for developing a trusted European cloud infrastructure [**?**]. In this context, the EU H2020 SecureCloud project (Secure Big Data Processing in Untrusted Cloud) proposes a innovative approach to ensure the dependability of critical applications that are executed in distributed, potentially untrusted cloud infrastructures. This innovative approach leverages the emergence of a new and promising technology – secure commodity CPUs – which promises to enable a new generation of dependable applications by basing trust in hardware mechanisms offered by commodity CPUs, in particular, Intel's Secure Guard eXtensions (SGX) [**?**]. This permits applications to be isolated not only from other applications in the cloud, but also from the underlying operating system and the hypervisor. It allows users to run their sensitive applications in a public cloud without the need to unconditionally trust the cloud provider. Moreover, to facilitatethe usage applications with high or very high security requirements, SecureCloud will integrate and extend the most popular technologies (i.e OpenStack, Secure Container, Coordination Service and Software-Defined Networks(SDN)) of last years to ensure the dependability of cloud applications. The paper is organised as follow: Section 2 provides a project overview, the proposed approach and its goals. In Section 3 the fundamentals of SGX technology is discussed. A real use case and an attack scenario through which will demonstrate the potentiality of SecureCloud approach are presented in Section 4. Finally, Section 5 concludes the paper with final remarks.

II. SecureCloud Overview

SecureCloud project focuses on a particularly important domain: applications that support critical infrastructures. If one can trust a cloud to run applications in the context of critical infrastructures, one can clearly trust this cloud to run applications in a large variety of application domains.

The area of interest of the project is the smart grids domain. Smart grid applications offer the opportunities to consider many of the requirements that a sensitive big data applications may have when executing in the cloud. First, smart grid applications deal with a growing volume of data. Smart meters and sensors for monitoring distribution and transmission grids are being deployed and are capable of continuously collecting and transmitting data. Adequate use of this data enables energy distributors not only to optimise their infrastructure, but also to reduce the environmental impact of supplying power to a given load or region. Second, these promising data analysis require to have access to detailed information about energy consumption. This data represents a big privacy risk because it can be used to accurately profile consumers activities and behaviors. Finally, when applications consider the data and decide to react to it, this reaction would interfere with the physical world. If an adversary compromises such actuating applications, the actuation could have devastating effects on the power system. In stark contrast to traditional throughput-oriented, batch-processing cloud applications, applications in the critical infrastructure domain do not only have strong requirements with respect to confidentiality, integrity, availability, but they typically are also latency sensitive. Secure storage of sensitive data in untrusted clouds is widely regarded as a solved problem [**?**] [**?**]. However, the secure and efficient processing of sensitive data in untrusted cloud is an open issue for securing cloud computing.
SecureCloud approach leverages and extends the following technologies:

- **Intel SGX** as root of application trust to provide confidentiality and integrity of sensitive data. SGX isolates the memory content of protected applications to prevent the operating system or the hypervisor from being able to read and/or modify application data.
- **OpenStack** as a common cloud stack infrastructure.
- **Container technology** to allow the execution of Intel SGX secure enclaves inside containers.
- **Coordination Service** to detect a computer or an application process failure and restart either the application process on a different computer or a newly created virtual machine, or the container depending on the requirements of the application process.
- **Software-Defined Networks(SDN)** to connect the application components within data centres as well as across data centres.

III. Intel SGX technology: basic concepts

The SecureCloud platform enables secure processing and the dependability of on-cloud critical applications exploit- ing the Intel's Secure Guard eXtensions (SGX) as key technology. SGX aims at solving the so called "secure remote computation" problem. This problem deals with the execution of applications on a remote host owned and maintained by an untrusted party. The solution proposed by Intel leverages trusted hardware (i.e. the Trusted Execution Environment (TEE)) in the remote computer. The SGX extensions are introduced with the 6th generation Intel core processors "Skylake" to protect selected code, data and sensitive processing from disclosure and modification. These extensions allow programs to allocate protected regions of execution in memory, known as secure enclaves, isolating application's secrets from external world and privileged

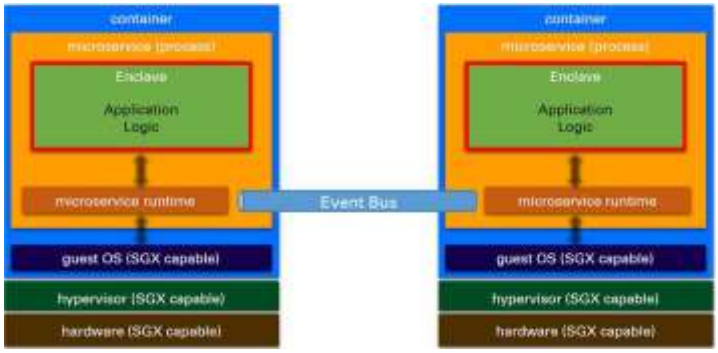

Figure 1. SecureCloud architecture

software, such as kernel and hypervisor. The access to the secure enclaves is regulated by security mechanisms enforced by the CPU. The information stored within an enclave (sensitive data and the code that processes it) are protected by the SGX that identifies and enables the applications that can execute it. Only trusted applications can access to the enclave, without having to trust the environment that hosted them. To do this, the SGX supports the software attestation feature which proves the reliability of a chunk of code running in an enclave. The attestation process convinces an enclave that it is communicating with another enclave that it is running in a secure container and also, that has not been altered. This mutual verification between the enclaves is enforced using processor key which is accessible only by a specific enclaves, called "Quoting Enclave". Given that, two applications could reside in the same or different machine and SGX supports both an intra-attestation and an inter-attestation procedure. The inter-attestation provides a secure channel between two remote enclaves by using a key exchange protocol, which the Diffie-Hellman protocol [**?**] is an example. Of course, the high security provided by the SGX technology has a cost both in performance (the verification of reliable enclaves has a non-negligible impact on the performance) and for the restricted set of operations that an application can do in an enclave (the system calls are not allowed). These limitations must be addressed by a careful design of the SGX-enabled applications, choosing appropriately what can be stored within the enclave and what can be outside of them.

IV. Smart Homes Application Use Case

As mentioned in Section **??**, in order to validate and demonstrate the need and the feasibility of using secure clouds, the SecureCloud project considers use cases in the area of smart grids. Smart grid applications offer the opportunities to consider many of the requirements that a sensitive big data applications may have when executing in the cloud. One of these use cases involves the collection of power consumption data from a residential user. The cloudification of smart home applications is a promising solution in terms of benefits for the society. However, migrating house energy management information to the cloud poses stringent security requirements to applications. Once this data is under control of an energy provider, an adversary could compromise this data, or a malicious employee could gain access to them. Data loss, data breach, data corruption, and Denial of Service (DoS) are the main risks to be considered if sensitive data are moved to a cloud-based infrastructure. Therefore, the data gathered from the on field sensors, especially their confidentiality, has to be protected. Also, a secure data storage and a reliable communication between smart meters and applications is necessary. Given the above requirements, is been defined the design of a Smart Homes application composed by several micro-services. Figure **??** shows the Smart Homes application micro-services and their integration with SecureCloud platform. In the following, an exhaustive list of the micro-services concerning the Smart Homes application is reported:

- **Data Collector** is the micro-service responsible for acquiring and forwarding data retrieved by the sensors toward other application micro-services.
- **Alarm Manager** is the micro-service responsible to spot alarms or critical conditions.
- **SQL Archive** to store historical data into a SQL database.
- **Access Controller** is the micro-service that performs the Identity Access Management (IAM) of the application users.
- **Web Proxy** is the micro-service that provides data coming from the Smart Homes application back-end to the front-end dashboard.
- **Dashboard GUI** is a web-based Human Machine Interface (HMI) for analysing historical data and detecting possible anomalous patterns. The GUI must establish a secure channel with the Web Proxy micro-service in order to protect the visualisation of the measurements.

Each micro-service will be executed in a distinct secure container and a secure event bus (i.e. ZeroMQ [**?**]) is been selected to guarantee a secure message exchange between

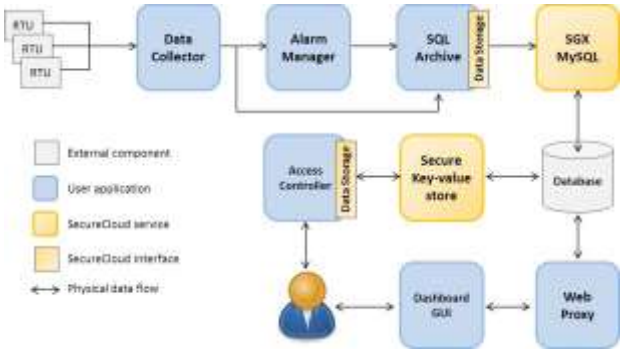

Figure 2. Smart Homes application micro-services

the micro-services.

A. Attack scenario

An application for acquiring, processing and storing data coming from a smart home represents an interesting target for a cyberattack. The interest dramatically increases when these applications use cloud-based infrastructure. This means that sensitive data, as well as its processing, is at risk. Therefore, the Smart Homes application use case will demonstrate how the SecureCloud approach mitigates this risk. Two fundamental security mechanisms will be shown. On one hand, the data acquired from the smart home sensors will be analysed in order to detect anomalous activities. The detection of an anomalous situation can be an indicator of a cyber or a real world attack. On the other hand, the system which executes this anomaly detection has to be secure itself. In order to show how the system can be implemented in a secure way the scenario focuses on a specific type of attack, named Iago attack [**?**]. The Iago attack consists in the mounting of a malicious kernel on the node where the target application runs. The malicious kernel runs at higher privileges on the processor, and can perform a cyberattack on the target application by reading application secrets from the memory or manipulating data. A malicious user can execute a Iago attack against the cloud machine where the anomaly detection application runs violating the integrity and confidentiality of data processed and stored in that machine. Security measures against Iago attacks have been investigated and presented in [**?**], where a paraverification technique that enforces the untrusted operating system to participate in its own verification is proposed. A different approach has been presented in [**?**], where a user-provided token returned by the operating system allows to verify that newly memory mapped regions do not overlap the old ones. The proposed attack scenario will demonstrate how the enhanced security technologies implemented in the SecureCloud platform thwart this type of threat. In particular, the SecureCloud SGX-enabled architecture will protect the memory contents stored in the enclaves from the malicious operating system and the use of ZeroMQ as event bus, combined with its security protocol, will prevent message breach during the communication among the enclaves.

V. Conclusion

SecureCloud will clearly make one step forward in the dependability of critical applications executed in potentially untrusted cloud infrastructures. To reach this meaningful goal, the SecureCloud approach combines secure CPU hardware in commodity processors and a trusted computing base. The integration of SecureCloud platform features into a standard cloud stack will encourage easy migration of critical, as well as non-critical applications, to the cloud without compromising application dependability. As a result, this project will contribute for influencing the standards and best practices to remove the barrier to the broad adoption of cloud computing in the critical infrastructure domain. The ease of adoption, in combination with the cost fairness of the cloud model, will enable enterprises or individual users, to create and deploy secure data-processing applications in the cloud.

Acknowledgments

SecureCloud has received funds from the European Union's Horizon 2020 research and innovation programme and was supported by the Swiss State Secretariat for Education, Research and Innovation (SERI) under grant agreement No 690111.

References

[1] "Cloud industry forum," https://www.cloudindustryforum.org/, accessed: 2017-10-04.

[2] "Trusted cloud europe," https://ec.europa.eu/digital-single-market/en/news/trusted-cloud-europe, accessed: 2017-10-05.

[3] V. Costan and S. Devadas, "Intel sgx explained." *IACR Cryptology ePrint Archive*, vol. 2016, p. 86, 2016.

[4] A. Bessani, M. Correia, B. Quaresma, F. André, and P. Sousa, "Depsky: dependable and secure storage in a cloud-of-clouds," *ACM Transactions on Storage (TOS)*, vol. 9, no. 4, p. 12, 2013.

[5] C. Wang, Q. Wang, K. Ren, N. Cao, and W. Lou, "Toward secure and dependable storage services in cloud computing," *IEEE transactions on Services Computing*, vol. 5, no. 2, pp. 220–232, 2012.

[6] A. S. Ibrahim, J. Hamlyn-Harris, and J. Grundy, "Emerging security challenges of cloud virtual infrastructure," *arXiv preprint arXiv:1612.09059*, 2016.

[7] A. López-Alt, E. Tromer, and V. Vaikuntanathan, "On-the-fly multiparty computation on the cloud via multikey fully homomorphic encryption," in *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*. ACM, 2012, pp. 1219–1234.

[8] M. Brenner, J. Wiebelitz, G. Von Voigt, and M. Smith, "Secret program execution in the cloud applying homomorphic encryption," in *Digital Ecosystems and Technologies Conference (DEST), 2011 Proceedings of the 5th IEEE International Conference on*. IEEE, 2011, pp. 114–119.

[9] J. G. Dyer, M. Lindemann, R. Perez, R. Sailer, L. Van Doorn, and S. W. Smith, "Building the ibm 4758 secure coprocessor," *Computer*, vol. 34, no. 10, pp. 57–66, 2001.

[10] A. Baumann, M. Peinado, and G. Hunt, "Shielding applications from an untrusted cloud with haven," *ACM Transactions on Computer Systems (TOCS)*, vol. 33, no. 3, p. 8, 2015.

[11] "What is docker?" https://www.docker.com/what-docker, accessed: 2017-10-06.

[12] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, 1976.

[13] P. Hintjens, *ZeroMQ: messaging for many applications*. " O'Reilly Media, Inc.", 2013.

[14] S. Checkoway and H. Shacham, *Iago attacks: Why the system call api is a bad untrusted rpc interface*. ACM, 2013, vol. 41, no. 1.

[15] O. S. Hofmann, S. Kim, A. M. Dunn, M. Z. Lee, and E. Witchel, "Inktag: Secure applications on an untrusted operating system," in *ACM SIGARCH Computer Architecture News*, vol. 41, no. 1. ACM, 2013, pp. 265–278.

[16] Y. Kwon, A. M. Dunn, M. Z. Lee, O. S. Hofmann, Y. Xu, and E. Witchel, "Sego: Pervasive trusted metadata for efficiently verified untrusted system services," in *ACM SIGPLAN Notices*, vol. 51, no. 4. ACM, 2016, pp. 277–290.