A Secure Cloud-Based SCADA Application: the Use Case of a Water Supply Network

Gianfranco Cerullo[1], Rosario Cristaldi[2], Giovanni Mazzeo[1], Gaetano Papale[1], and Luigi Sgaglione[1]

[1] University of Naples 'Parthenope', IT,
{gianfranco.cerullo,giovanni.mazzeo,
gaetano.papale,luigi.sgaglione}@uniparthenope.it
[2] EPSILON srl, IT,
rosario.cristaldi@epsilonline.it

Abstract. Cloud computing paradigm is gaining more and more momentum, to the extent that it is no more confined to its initial application domains, i.e. use by enterprises and businesses willing to lower costs or to increase computing capacity in a flexible manner. In particular, increasing interest is recently being paid to the huge potentials - in terms of benefits for the society at large - that might result from the adoption of cloud computing technology by critical infrastructure (CI) operators. This is of course putting special emphasis on the need for dependable and trustworthy security mechanisms in cloud technology based services, since a critical infrastructure is vital for essential functioning of a country. Incidental or deliberate damages to a CI have serious impacts on the economy, and possibly make essential services unavailable to the communities it serves. In this paper we present the proof-of concept of a cloud-based Water Supply Network Monitoring (WSNM) application, named RiskBuster (RB), that ensures the confidentiality and integrity of SCADA monitoring data collected from dam sensors and stored in the cloud by using the innovative Intel Software Guard eXtension (SGX) technology.

1 Introduction

The industrial community is now aware of the tremendous advantages coming from the adoption of Cloud technologies. The outsourcing to the Cloud is a well-consolidated trend in many fields, which has a unique barrier: the exposure of sensitive data to security risks. While some companies have overcome this fear by relying in the efforts spent by academic and industrial communities to make the cloud more secure, others are still skeptical due to the increasing numbers of attacks occurred in the last years.
In this sense, a glaring example comes from the Critical Infrastructure (CI) sector, an essential branch of societies in terms of economic viability and livability, spanning all countries' fundamental facilities such as energy, telecommunications, water supply, transport, finance, and health. Through a Cloud-based Supervisory Control And Data Acquisition (SCADA) system, providers and users can

dramatically reduce costs eliminating the expenses related to the installation and maintenance of IT infrastructures, enhance functionalities, and achieve greater reliability.
However, CIs do not undertake to the Cloud Computing (CC) hampered by security concerns. Unlike other sectors, in fact, CIs have more stringent security requirements due to the terrible impact that attacks directed to different targets including assets, networks, and systems may have on Nations' safety, prosperity, and well-being. Evidences as the recent "Black Energy 3" attack enforced in Ukraine [1] in 2015 or the "Havex" SCADA-focused trojan launched in 2014 [2], demonstrate how important is the security of Industrial Control Systems (ICS). The adoption of cloud for critical applications needs new approaches and methods. Emerging technologies have the potential to meet the demands of security coming from CIs organizations.
In this work we propose RiskBuster (RB), a proof-of-concept of a cloud-based application for a Water Supply Network Monitoring (WSNM) use case. Unlike traditional applications RiskBuster is designed to make use of advanced security mechanisms provided in the context of the Secure Enclaves for REactive Cloud Applications (SERECA) project[3]. In particular, RB aims to protect the confidentiality and integrity of sensitive monitoring data stored in the cloud using Intel Software Guard eXtension (SGX), the new extension of Intel Instruction-Set Architecture (ISA) provided in newest Skylake processors.
SGX permits to protect application state from the hypervisor and the operating system. Since all data is encrypted in memory in *Secure Enclaves*, and only the CPU has access to the encryption keys, even physical access to a machine does not help to gain access to data protected with SGX.
The incorporation of secure enclaves into a cloud computing software stack could allow an extremely high level of security in RiskBuster, which makes the adoption of cloud much more reliable for the Critical Infrastructures. We believe that our work gives a significant contribution in this sense.
The remainder of this paper is organized as follows. Section 2 reports other works moving in our same direction. Section 3 analyzes the type of threats that may arise in a cloud environment. Section 4 describes the WSNM use case by also classifying the type of requirements. Section 5 provides an overview of the monitoring application by reporting the technological choices and how we conceive the integration with SGX to counter security threats. Finally, section 6 concludes the document.

2 Related Work

Protecting CIs in cloud and non-cloud platforms is of main concern [9][7][6][3]. The process of critical infrastructures cloudification is testified most of all in the field of SmartGrids. Such infrastructures are particularly worrying in terms

[3] A European Project H2020 funded in the context of ICT-07-2014: Advanced Cloud Infrastructures and Services. Grant agreement no: 645011

of security as depicted in [10]. Several works discuss the advantages in leveraging cloud techniques to meet the demand for increasingly intelligent and cost-effective systems. In this regard, Bera et al. [3] provided a thorough survey in which they analyzed the impact on SmartGrids, positive and non, of a migration to the cloud. In particular the discussion was focused on three main aspects: energy management, information management and security. It is precisely the last one that is left as an open issue by the authors due to the many existent deficiencies of current protection systems. Therefore, it is not surprising that they reported a huge amount of researches related to possible security mechanisms in SmartGrid applications. However, none of these rely on innovative solutions that can address the extraordinary security requirements of CIs.

An approach for a secure Cloud-based monitoring system was proposed by Sule et al. [9], which in their work describe a secure cloud deployment for mission critical applications in the energy sector. Their solution was based on the usage of Trusted Platform Modules (TPM) to eliminate the need of custom software and patches by enforcing hardware integrity measurement and testing through a cryptoprocessor provided in cloud providers' nodes. Such a technique produces a Static Root of Trust Measurement (SRTM) useful to verify all the software loaded since BIOS. Moreover, to enhance their work, the authors showed how this solution affects the performances declaring an additional overhead caused by the cryptoprocessor of 2.65 second. The mechanism employed is on the same vein of our solution proposed, however Intel SGX does much more. It builds a Trusted Execution Environment (TEE) that allows a Dynamic Root of Trust Measurement (DRTM). SGX no longer depends on the TPM to enforce measurements, sealing and attestation. Instead there is a special enclave that emulates the TPM. There is no more an external co-processor, everything runs on main CPU so is much faster.

In the same way, Baker et al. [6] presented a concept for a cloud platform to reinforce the integrity and security of SOA-based SCADA systems exploited in the context of Critical Infrastructures. The paper makes use of requirements imposed by a SmartGrid CI to highlight the applicability of the proposed platform in a real world scenario. They analyzed a collection of existing measures (e.g. Intrusion Detection Systems (IDS), Role-Based Access Control (RBAC)) to define a 'Security Toolbox' that could be adopted to guarantee an acceptable level of security. Furthermore, they proposed a Multilevel User Access Control Layer (MLAC) used to connecting users to the most appropriate SCADA node according to their role and credentials. This work can only give a partial contribution to address security issues, in fact, a more complex threat model in which, e.g, cloud administrators are malicious would be unmanageable.

3 Threats to CIs in a Cloud Environment

Although the migration of CIs' IT to the cloud lays the foundation for advanced cyber infrastructures - bringing with them a greater resistance to trivial attacks and, most of all, an improvement of reliability - a number of new security threats come out. Data loss or compromise, loss of organizational control, account hijacking and Denial of Service (DoS) are the noteworthy risks that organizations must take into account after proceeding to a cloud based IT infrastructure. The research community is proposing new security solutions as testified in [5] where authors survey current available security mechanisms in cloud infrastructures to address most relevant open issues. However, it is still not enough. The adaptation of Cloud-hosted SCADA systems to the cloud migration is paramount due to the lack of appropriate security mechanisms in most used SCADA protocols such as Modbus and DNP3 which do not support or perform authentication and encryption.

According to NIST, cloud computing "presents certain unique security challenges resulting from the cloud's very high degree of outsourcing, dependence on networks, sharing (multi-tenancy), and scale" [4]. Fernandes et al. [8] provide a survey of research literature to highlight cloud security open issues and challenges. Some of them particularly concern the context of Industrial Control Systems (ICS). We report here the most relevant ones:

- T1: Data Breach - CIs assets' data is the main target of malicious intrusive actions. Data confidentiality is certainly important but it is our belief that data integrity is much more important because attacks (e.g., manipulating sensor or control data) on the SCADA system could have terrible effects on nearby population safety as it misleads operators into making wrong decisions. Just think the impact of false data information on water quality in a distribution network.

  The problem of data outsourcing in a cloud migration is of immediate concern.
- T2: Account or Service Traffic Hijacking - The attempt to steal the access of operators is one way through which hackers can enforce malicious operations against the CI. The intruder can get into critical areas of a deployed monitoring service and possibly compromise the confidentiality, integrity, and availability of those services.
- T3: Denial of Service (DoS) - The high availability of SCADA systems is a fundamental requirement. Precisely for this reason, attackers may aim to an outage of CIs through a DoS or Distributed DoS (DDoS) attack. This attack is more dangerous in a cloud environment, since when the workload increases with respect to a specific service, the cloud environment provides additional computational power to that service. This means that on the one hand the cloud system counters the effects of the attack, but on the other hand it supports the attacker in his evil activity, by providing him with more resources.
- T4: Shared Technologies Vulnerabilities - The multi-tenancy feature of cloud technologies is extremely risky in terms of security if the hypervisor is not well secured. An evil activity on CI data integrity and confidentiality may be enforced through penetrations in virtual machines (VM) residing on the same hypervisor.
- T5: Malicious Insiders - Malicious administrators of the Cloud Provider (CP) or any other system administrator with privileged access to resources are a

consistent threat that traditional security mechanisms hardly are able to cope.

4 The Water Supply Network Use Case

A civil Water Supply Network (WSN) infrastructure has been chosen as a use case to validate and demonstrate our work. Such an infrastructure is extremely representative because it is composed by assets like network pipes, dams, and basins that make it a safety-critical infrastructure with strong security requirements.

Our WSN use case is under the administration of a public authority namely *Ente per lo Sviluppo dell'Irrigazione e la Trasformazione Fondiaria in Puglia, Lucania ed Irpinia (EIPLI)*, which provides water to a large part of Southern Italy population. EIPLI is in charge of the distribution of water - around 600 million cubic meters per year - for different uses (e.g. industrial, energetic, irrigation, drinking) to the nearest population. Besides the management of the water pipelines, the organization is also responsible of 8 dams with different storage capacities up to 550Mmc.

A cloud-based real-time monitoring application is an attractive solution for WSN administrators because they can discern from the burden of IT systems managements costs and easily interconnect all the assets possessed, which are geographically distant between each other. Furthermore, except for the aspects of security, the choice of adopting cloud technologies is further strengthened by the requirements imposed (functional and non-functional) that we are going to define in the following subsections.

4.1 Functional Requirements

Generally speaking, main monitoring functionalities needed by the WSN application are: the water quality monitoring and control (e.g. water turbidity) in order to avoid water pollution situations that could be disastrous for the population; infrastructure integrity monitoring to ensure that there are no structural problems in any part of the infrastructure (e.g. collection points, dams, tanks, valves, and pipes); and other key parameters monitoring, particularly, those with a direct impact on the operation of the water network (e.g. pipe pressure).

More specifically, the functionalities needed by the application are those typical of a SCADA system. That is:

- FR1 - Sensor Data Acquisition - It is needed the data acquisition from sensors, meters and field devices, such as photo, pressure, temperature and flow sensors needed to monitor fundamental parameters for the water quality and the infrastructure integrity.
- FR2 - Sensor Data Collection, storage and retrieval - The enormous amount of data acquired by the sensors must be handled and stored for subsequent analysis such as statistical trends, regulations, future load planning or billing.

- FR3 - Event and Alarms Processing - Any suspicious activity should be reported by the WSNM application. An Alarm Management System (AMS) must clearly show appropriate alarms indicating abnormal situations, varying priority and consequences, and keeping history of all the alarms generated. It must react with alarms of different level of importance in case sensors measurements go out of certain thresholds. A correlation of events of different nature should be enforced to produce representative and complex alarms.
- FR4 - Monitoring through Human Machine Interface (HMI) - Operators needs to interface with the system to monitor in real-time the status of the WSN infrastructure and react to eventual alerts or alarms. They also need to have access to the historic data storages to make appropriate assessments.

4.2 Para-Functional Requirements

The WSN is unarguably exposed to risks that are related to natural phenomena, disasters, and criminal/terrorist activity. For this reason, the non-functional requirements imposed, especially those related to the security and dependability, are particularly stringent. These are the cause of the skepticism to promote a serious migration of CI applications to the cloud. The monitoring application should guarantee the following:

- NFR1 - Data Security - Critical decisions are usually taken using the measurement data provided by the monitoring system. This means that data confidentiality and, especially data integrity are of primary importance. The WSN application should enforce mechanisms able to protect the data from malicious accesses and modifications.
- NFR2 - Service Availability and Responsiveness - A timely response to problems in the WSN is an important requirement, e.g., to have operators react quickly to critical conditions such as a terrorist attack on the WSN. Furthermore, it is equally important the continuous service provision. An outage could be fatal for the population.
- NFR3 - Service isolation - As a CI, any system associated with the WSN must comply with a variety of regulations and standards. Validation of compliance is a complex and expensive process though, especially when third-party CPs are involved.
- NFR4 - Scalability - The WSNM application must tolerate load peaks of the processing throughput. This capability must be maintained also for an increased number of feeding components (i.e. more sensors or event analysers).
- NFR5 - Interoperability - The application must support an easy integration of technologies of different nature and also allow the usage of software modules written in different programming languages.

5 The WSNM Cloud-Based Application

In this section we provide the proof-of-concept design of the proposed cloud application. We assume that the *Infrastructure-as-a-Service (IaaS)* Cloud Provider gives support for SGX-equipped processors and that in a *Platform-as-a-Service (PaaS)* delivery are available standard databases (specifically *Key/Value* storage) and coordination services.

In a first stage we present a general architecture of the WSN cloud-based monitoring application showing how the data flows from the WSN to the cloud. Then, we go further in the discussion analyzing the approach that could be followed to enhancing security of our SCADA application. Finally, we survey the technologies that may be used to realize the proposed application.

5.1 General Architecture Overview

Figure 1 reports the general architecture of our WSN cloud-based monitoring application. On the infrastructure side, sensors deployed around the WSN are responsible for retrieving measurements related to specific physical phenomena; their measurement signals are converted to digital data by means of signals converters components, known as Remote Terminal Unit (RTU).

The data acquired by the sensors is then transmitted to a *Data Communication Layer (DCL)*. This consists of "Data Logger" equipment, which allow secure communication between RTUs and the control/monitoring system by means of heterogeneous data communication layers, adopting different physical communication media and technologies (i.e., wired and wireless). Multiple equipment are deployed around the WSN, each one is responsible of a group of sensors installed nearby.

Subsequently, all the sensors data collected by the DCLs is given to *Gateway* machines in charge of preprocessing and then transmitting the sensor data acquired to the cloud platform. A gateway acts as a bridge between the WSN critical infrastructure and the cloud by providing interfaces for two communication protocols, one for the infrastructure and one for the cloud side.

The acquired data coming from the WSN is sent to the *Message Broker layer*, the mean through which the different Services (S) communicate at cloud-level. The broker distributes data, messages and events to all the units involved, in charge of specific functionalities, e.g, data storage management, data computation, data provision and alarms management. Besides the storage of the business critical data, both batch and real-time processing needed to make historical and real-time parameters calculations leverage the highly scalable compute capacity offered by the cloud platform. Except for the data acquisition, each feature of the WSN application is supposed to be executed at cloud-level.

5.2 An Innovative Approach to Secure Data in a Cloud Environment

Malicious attacks to the sensor data of the CIs SCADA system residing in the cloud are the main concern. The security open issues, listed in 3, deserve an innovative solution that may be able to give guarantees and trust even when a malicious insider with administrator privileges tries to manipulate sensitive data. A solution that may succeed in what the homomorphic encryption failed,

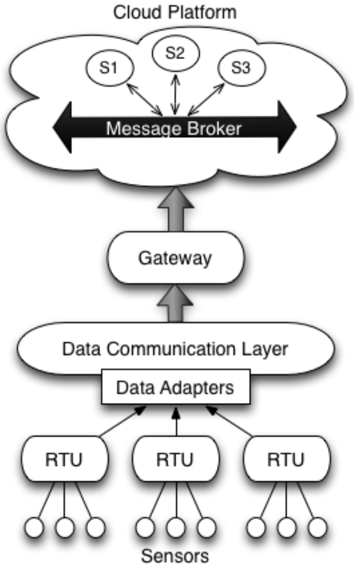

Fig. 1. WSN Monitoring Application General Scheme

e.g., the computation efficiency.
Therefore, we propose the usage of a novel technology recently released by Intel, namely Software Guard eXtension (SGX) [12], which is gaining more and more momentum in the research community, to harden our SCADA application.

Fundamentals of Intel SGX Intel SGX can be conceived as a ”Reverse Sandbox”, which aims to protect the integrity and confidentiality of code and data by creating a reserved memory space known as *enclave* able to protect an application from the external environment (including the OS or the Hypervisor). The enclave contains the sensitive data and the code that uses it. By doing so, SGX enables users to enforce the security of their code and data without having to trust the cloud provider.
Key point of SGX is the *remote software attestation* [11] feature useful to prove the goodness of a piece of software running in an enclave. Such a process convinces an enclave that it is communicating with another enclave that has a specific measurement hash, that is running in a secure environment and that has not been modified. The mutual verification between the enclaves is enforced by using a processor key, which is accessible only by a special enclave known as *Quoting Enclave*.
SGX supports both an intra-attestation and an inter-attestation service, that is, the SGX-enabled procedure of enclave identity verification can be empowered between two enclaves residing on the same host or between two enclaves residing

in different hosts. The remote attestation service builds a secure channel between the two enclaves by performing a Diffie-Hellman key exchange.

Designing a SGX-Enabled Application Hardening the monitoring application with Intel SGX is challenging. A first question that may arise is: how to design the application using secure enclaves. An example of an approach has been proposed by Baumann et al. [13] that included in the enclaves *LibOS*, a set of libraries able to run on a dedicated minimal kernel API surface. While this solution has the advantage of allowing the execution of generic applications with SGX, it has the drawback of a too big Trusted Computing Base (TCB) into the enclaves that could lead to security leaks.

Another possibility is to properly partition an application in order to keep the TCB as small as possible. Certainly, it is a non-trivial approach as it is quite difficult to figure out how to define the partitions. Such a solution was adopted by Schuster et al. [14] that partitioned a Hadoop MapReduce framework in order to enforce secure data analytics using SGX.

Finally, a third approach, the one we propose to adopt, is to use microservices into the enclave leveraging the intrinsic properties of microservices, that is, a framework with already well-partitioned functionalities. The only drawback coming from this solution is that the developer needs to re-engineer the application in order to define the microservices that need to be put into the enclaves. However, in the context of CIs this is acceptable because many industries require a dedicated application which already embeds a re-engineering process.

Because the microservice frameworks usually rely on other external services (e.g. databases, coordination services), it is equally important to ensure the execution of their security-critical functionalities inside the enclaves to guarantee a 360°protection. A method for partitioning the trusted and the untrusted parts of these services should be found as protecting the *data at rest* and the *data in transit* is also of concern.

5.3 Implementation Details

Starting from the requirements imposed by the WSN use case and from the design approach requirements needed to use SGX, we revised the architecture of a typical SCADA system to create a secure reactive cloud application. These growing class of applications, widely used in the Internet of Things (IoT) world, are typically highly interactive, scalable, resilient, responsive and event driven; a set of characteristics perfectly in line with the application Non-Functional requirements defined in 4.2.

Following a well-consolidated trend in the cloud world, we propose a SCADA application with a *microservice* architecture. This software pattern allows applications to be composed of modular, independent, autonomous, self-contained service units.

The glue between the microservices is needed: an open source framework known as *vert.x*[4] can do that, it is the base for distributed reactive application based on microservices. Vert.x is a polyglot event-driven application framework, which abstracts from multi-threaded programming allowing to write thread-safe concurrent applications as single-threaded ones. It is platform-independent and employs an asynchronous, actor-like programming model using event handlers. Its nervous system is the highly scalable *EventBus* which allows the communication of the microservices units through different type of messaging patterns (e.g. publish/subscribe, point to point). In SCADA case, the preferred path is a *Publish/Subscribe* communication pattern: each sensor data is published on the EventBus by the *Gateway* and all the interested microservices subscribe themselves to the topic (or the sensor) of interest.

The WSN application is composed by different microservices, namely *verticles* in *vert.x* jargon, that interact between them through the EventBus. Each verticle represents a microservice responsible of an independent process. Thus, in our application most important are: the "Data Collector" (or Data Publisher) which acquires data from the sensors and transmit everything to the vert.x EventBus responsible of sorting it to the subscribed verticles; the "Access Controller" which enforces authentication and authorization policies to allow the access to the resources; the "Alarm Manager" which realize the stream processing to signal alarms or critical conditions; the "Archiver" responsible of storing Real-Time and Historical data into the correspondent databases; the "Web Proxy" which acts as medium between users connected to the dashboard and application's resources; and finally the "Dashboard GUI" which provides a web-based Human Machine Interface (HMI) to monitor sensors' status.

All the above mentioned services should run partially inside of enclaves and partially outside enclaves. The goal is to protect the usage of the private datasets into secure enclaves and, equally important, to enforce the communication of these data through secure enclaves channel. Leveraging the remote attestation feature of SGX can help in this sense. The on-site gateway and the broker on the cloud platform attest each other identities establishing in this way a secure channel. The data will be exchanged only between enclaves and will never be decrypted out of them.

To do that, a SGX-enabled version of vert.x must be provided. Vert.x applications runs on top of the Java Virtual Machine (JVM). Extending vert.x means including, first, the JVM into the *secure enclaves* of SGX in order to enable the usage of SGX-aware security-critical functionalities of vert.x. The inclusion of vert.x into a SGX enclave can not leave aside the JVM but the Oracle's HotSpot would be unfeasible. A lighter version of the JVM is desirable, the *JamVM*[5] represents a good compromise even though lacks the Just-In-Time (JIT) compiler, which means a degradation of performances.

The SCADA application requires persistent storage of the time series data generated by the data acquisition equipment. All data in the database should be encrypted at rest and only be accessible to clients holding the decryption key.

[4] http://vertx.io

[5] `http://jamvm.sourceforge.net`

For what about the databases, *MongoDB* represents a good choice. This suits really good for its *key-value* intrinsic characteristic. In fact, the problem of store and retrieve big amount of data at a high rate is well addressed by Mongo through the typical key-value storage system which collides perfectly with the requirements of a real-time monitoring system characterized by time series data. In order to optimize read/write operations, the data stored is organized as follows: each minute represents a Mongo document and samples of the specific minute (60 samples, one per second) are stored inside it.

A secure version of the key-value storage system needs that the data is encrypted before being stored with a specific key and that data can only be decrypted by the owner of the key. An extension of Mongo should include new storage functions in which the encryption and the decryption, transparently to the user, happens into SGX enclaves.

Another noteworthy architectural element is the Alarms Management System that meets requirement FR3. We propose *EsperTech CEP*[6]. This is a Complex Event Processing (CEP) system that identifies and analyzes cause-and-effect relationships among events in real time. Every abnormal situation generates events interpreted by the CEP that, based on the rules defined by the administrator, enforces a correlation process and outputs a single alarm level. What is extremely sensitive for the CEP security is the event correlation. Such a process needs to be enforced in the enclaves to ensure that the monitoring events won't be modified.

6 Conclusion

In this paper we described the approach conducted to design a secure cloud application for a critical infrastructure using new Intel ISA extension known as SGX. We highlighted the positive effects of a migration of critical infrastructures IT to a cloud environment but also what this may result in terms of security. Then, we presented a proof-of-concept of a SGX-enabled SCADA application that may address some open issues that are now hampering the adoption of cloud technologies in the field of CIs. Our work was strengthened through a real use case consisting in a water supply network that provided requirements extremely useful for the analysis.

Acknowledgments

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 645011.

References

1. Attack to ICS: Black Energy `https://ics-cert.us-cert.gov/alerts/ICS-ALERT-14-281-01B`

[6] `http://www.espertech.com`

2. Attack to ICS: Havex `https://ics-cert.us-cert.gov/alerts/ICS-ALERT-14-176-02A`
3. S. Bera, S. Misra and J. J. P. C. Rodrigues, "Cloud Computing Applications for Smart Grid: A Survey," in IEEE Transactions on Parallel and Distributed Systems, vol. 26, no. 5, pp. 1477-1494, May 1 2015. doi:10.1109/TPDS.2014.2321378
4. Timothy Grance; Wayne Jansen; "Guidelines on Security and Privacy in Public Cloud Computing"; NIST
5. Luigi Coppolino, Salvatore DAntonio, Giovanni Mazzeo, Luigi Romano, Cloud security: Emerging threats and current solutions, Computers & Electrical Engineering, ISSN 0045-7906, http://dx.doi.org/10.1016/j.compeleceng.2016.03.004. Keywords: Cloud computing security; Security techniques; Intel SGX; Homomorphic cryptography; Cloud platforms
6. T. Baker, M. Mackay, A. Shaheed and B. Aldawsari, "Security-Oriented Cloud Platform for SOA-Based SCADA"; Cluster, Cloud and Grid Computing (CCGrid), 2015 15th IEEE/ACM International Symposium on, Shenzhen, 2015, pp. 961-970. doi: 10.1109/CCGrid.2015.37
7. Coppolino, L., D'Antonio, S., Formicola, V., Romano, L. Integration of a system for critical infrastructure protection with the OSSIM SIEM platform: A dam case study (2011) Lecture Notes in Computer Science, 6894 LNCS, pp. 199-212.
8. Diogo A. Fernandes, Liliana F. Soares, Joo V. Gomes, Mrio M. Freire, and Pedro R. Incio. 2014. "Security issues in cloud environments: a survey" Int. J. Inf. Secur. 13, 2 (April 2014), 113-170.
9. M. J. Sule, M. Li, G. A. Taylor and S. Furber, "Deploying trusted cloud computing for data intensive power system applications" Power Engineering Conference (UPEC), 2015 50th International Universities, Stoke on Trent, 2015, pp. 1-5. doi: 10.1109/UPEC.2015.7339864
10. Coppolino, L., D'Antonio, S., Romano, L. Exposing vulnerabilities in electric power grids: An experimental approach (2014) International Journal of Critical Infrastructure Protection, 7 (1), pp. 51-60.
11. Ittai Anati, Shay Gueron, Simon P Johnson, Vincent R Scarlata. "Innovative Technology for CPU Based Attestation and Sealing". In Proceedings of the 2nd International Workshop on Hardware and Architectural Support for Security and Privacy, HASP, volume 13, 2013.
12. Frank McKeen, Ilya Alexandrovich, Alex Berenzon, Carlos Rozas, Hisham Shafi, Vedvyas Shanbhogue, and Uday Savagaonkar. "Innovative Instructions and Software Model for Isolated Execution" HASP, 13:10, 2013.
13. Andrew Baumann, Marcus Peinado, and Galen Hunt. 2015. "Shielding Applications from an Untrusted Cloud with Haven". ACM Trans. Comput. Syst. 33, 3, Article 8 (August 2015), 26 pages. DOI=http://dx.doi.org/10.1145/2799647
14. F. Schuster, M. Costa, C. Fournet, C. Gkantsidis, M. Peinado, G. Mainar-Ruiz, and M. Russinovich. "VC3: trustworthy data analytics in the cloud using SGX". In 2015 IEEE Symposium on Security and Privacy, SP 2015, San Jose, CA, USA, May 17–21, 2015, pages 38–54, 2015.