A Secure Self-Identification Mechanism for Enabling IoT Devices to Join Cloud Computing

Massimo Villari(✉), Antonio Celesti, Maria Fazio, and Antonio Puliafito

University of Messina, DICIEAMA,
Contrada Di Dio (S. Agata), 98166 Messina, Italy
{mvillari,acelesti,mfazio,apuliafito}@unime.it
http://mdslab.unime.it

Abstract. Nowadays, one of the major problems in Internet of Things (IoT) is the initial setup and boot up of new embedded devices that have to be connected over the Internet. On the other hand, another problem is the interaction of such devices with a Cloud computing environment. This paper deals with the possibility to automatically configure IoT devices in a secure way, so as to provide new added-value services. In particular, after a discussion of a Cloud scenario for IoT, we discuss how to perform a self-identification process in order to achieve a secure auto-configuration of IoT devices joining the Cloud. The paper deals with the design of secure IoT infrastructures.

Keywords: Cloud computing · IoT · Identification · Configuration · Security

1 Introduction

Nowadays, the increasingly penetration of sensing devices and the emerging concept Internet of Things (IoT) offer new possibilities for sharing data and services over the Internet. As highlighted in the *Digital Agenda for Europe* [1], one of the key challenges for the European Commission is to have a globally competitive Cloud infrastructure for the "Internet of Services" interconnected with "Things" distributed over remote areas. IoT is currently applied in many applications fields, such as in buildings construction, car traffic monitoring, environment analysis, health-care, weather forecast, video surveillances, etc. Definitely, there is not limit to the possible scenarios that can be accomplished combining IoT and Cloud computing. In our opinion, IoT can appear as a natural extension of Cloud computing, in which the Cloud allows us to access IoT based resources and capabilities, to manage intelligent pervasive environments. In addition Cloud computing can support the delivery of IoT services. Thus an IoT service can be considered as an on-demand Cloud-based Sensing and Actuation as a Service (SAaaS). One of the main problems in deploying IoT devices is the secure self-configuration of such devices that is necessary to interconnect them over the Cloud. In this paper, we analyze the existing issues regarding to the self-configuration of IoT devices that have to be connected over the Internet to join

R. Giaffreda et al. (Eds.): IoT360 2014, Part II, LNICST 151, pp. 306–311, 2015.
DOI: 10.1007/978-3-319-19743-2_41

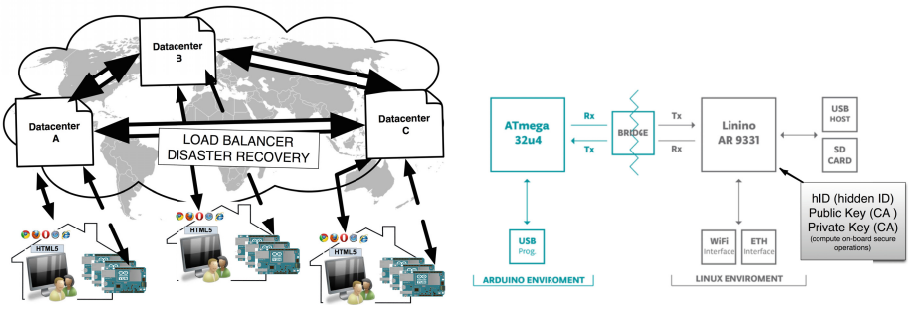
a Cloud environment. In our opinion, an IoT device should be able to configure itself, interacting with the Cloud in a secure way, and downloading its customized features directly from remote providers. The paper we present hereby is strongly related to the design of secure IoT infrastructures. According to our vision, each user should be able to turn on his/her IoT device, connecting it via WiFi, and waiting for its self-configuration in order to interact with the Cloud. In order to self-configure IoT devices in a secure way and allowing them to interact over the Cloud, they should be equipped with capabilities including security keys, cryptographic algorithms, hidden IDs, etc. The rest of the paper is organized as follows. Section 2 discusses related works. Section 3 presents a possible scenario integrating IoT and Cloud computing. In Sect. 4, we highlight the main factors involved for a secure self-identification of IoT devices. In Sect. 5, we discuss how IoT devices joining a Cloud system can self-register themselves to perform a self-configuration process. Section 6 concludes the paper.

2 Related Work

In the near future, the heavy penetration of sensing devices into Internet applications will cause the explosion of the amount of data to be stored and processed, as very well described in [10]. Often Sensor Networks are considered as virtual devices [7]. Physical sensors can be mapped into virtual sensors clouds, hence Cloud Computing and Sensors Networks can be managed in the same way. Sensor Network and IoTs also reside a lot of problem in security context as described in [5,11] and [12]. The new concept of Mobile Cloud computing appeared in 2012. The concept consists in providing new services for Cloud users taking into account their movements and preferences. In [4], the authors investigated the delivery of mobile cloud services. They stated that services suffering from poor performance due to the mobile network fluctuations. Moreover, under the Smart Cities umbrella many works are dealing with sensors and Clouds, as well as in [9] and [8]. In [9], the authors have investigated the possibility to unify Resilient Cloud Computing and Secure IoT in smart cities scenarios. Security and resilience seen to look at the same perspective. In the same direction respect to our work is the work described in [6]. Here, the authors present an exhaustive analysis of sensor Cloud architectures benefiting of Arduino devices. Very similar is the study presented in [3] in which the service provisioning for sensors is assessed leveraging Clouds.

3 A Cloud Scenario for IoT

In this Section, we present a scenario in which several IoT devices interact with a Cloud system. Figure 1(a) shows different users holding several IoT devices connected to a domestic WiFi network. Each device is able to automatically configure it-self downloading its configuration from a given Cloud provider. Several datacenters belonging to a Cloud operator are spread over the world. For example, datacenter A is placed in USA, datacenter B is located in Europe, and



(a) Single Cloud Scenario with one operator distributed among datacenters that interact with IoT devices and Customers.

(b) Arduino Yun extended with security capabilities.

Fig. 1. The Cloud Scenario under investigation and the modified version of Arduino Yun.

datacenter C is placed in Asia. Each datacenter collects data coming from IoT devices connected in geographical area that it serves.

An indispensable requirement of the proposed scenario is that when an IoT device moves from a datacenter to another, it should be able to self-configure itself in a secure way.

4 Towards Secure Self-Identification of IoT Devices

This Section, we discuss how existing IoT embedded devices might be extended and used to achieve the scenario previously described. According to the approach discussed in this paper, these IoT devices should be onboarded with Security Keys, Cryptographic Algorithms and Hidden IDs (hIDs).

4.1 Arduino Yun

In order simplify the discussion on how an IoT device can be extended with hardware security capabilities we consider Arduino Yun as reference. The Arduino open hardware framework is a consolidated architecture able to fulfill the IoT requirements especially for its cheapness and simplicity of utilization. Many versions, shields, and extensions exist over the market for the Arduino platform. One of these versions is the new framework able to provide the Arduino capabilities along with the Linux Embedded features that is referred as Yun device. Specifically, the Yun is distinguished from other Arduino boards by the fact that it can communicate with the Linux distribution onboard, offering a powerful networked computer with the easiness of Arduino. The Atheros AR9331 processor supports a Linux distribution based on OpenWRT named Linino. Figure 1(b) shows how Yun works. The left side of the picture depicts the Arduino part, whereas in the right side there is the Linino part. Yun has a built-in WiFi/Ethernet boards that enrich the Arduino part. The Linux embedded part can be used for accomplishing the interactions with the Cloud.

4.2 Security Keys, Cryptographic Algorithms and Hidden IDs

In order to achieve the scenarios discussed in Sect. 3, an IoT device such as Arduino Yun should be equipped with a component mounted on the board during the manufacturing and offering several security capabilities as depicted in the right part of Fig. 1(b). In particular, these security capabilities should include: Security Keys (e.g., a couple of public/private keys X509v3 based (K_{pub}, K_{priv})), Cryptographic Algorithms, a hidden ID (hId). The *hID* is a numeric serial-number used by manufacturer for recognizing each board. It is hidden because no one must read it. Here, we introduced the concept of Obfuscated ID (obH) derived from the MD5 hashing function. The major property of an hashing function is its incontrovertibly, in fact it is also defined an one-way function (i.e., from the output of an hashing function it is not possible to deduct the input). Hence, looking at Eq. 1 the obH is useful for tracking the board without knowing its public MAC address and the board owner if the MAC-User association exists in the Cloud provider.

$$obH = hash(hID, MAC) \tag{1}$$

The obH is computed by Eq. 3, and it represents a board index that can be stored in whichever public database. According to Eq. 2, in any communication between the device and the Cloud operator a Message (M) can be included in the body of all communications concatenating obH, MAC, and a public key K_{pub}.

$$M = concat(obH, MAC, K_{pub}) \tag{2}$$

A signature strongly guarantees the trustiness of the sender. The public key K_{pub} is signed at production level during its manufacturing, using for example the Certification Authority (CA) of the manufacturer of the IoT device. Instead, the private key K_{priv} is not accessible externally from its chip endorsed in the device, but it can be used by the internal security algorithms.

$$SM = signature(K_{priv}, M) \tag{3}$$

4.3 Adding Secure Hardware Capabilities

Trusted Computing (TC), defined by the Trusted Computing Group (TCG) [2], combines hardware and software security mechanisms to enhance the security level of computing environments. TC implies the adoption of an hardware chip called Trusted Platform Module (TPM), that is able to provide Roots of Trusts (RoTs) and to extend its trust to other parts of the device by building a chain of trust. It offers facilities for the secure generation of cryptographic keys by means of a unique RSA key burnt into as it is produced (i.e., the Endorsement Key (EK)). The TPM includes capabilities such as machine authentication, hardware encryption, signing, secure key storage and attestation. Born for securing traditional Personal Computers, the TCG is currently looking at both embedded and mobile devices whose reference architecture specification drafts were released respectively in April and June 2014. The specifications provide guidelines on

how to onboard the TPM in a device even though there have not been so many implementations yet on real hardware devices. TC and embedded systems are at the early stage, however, in our opinion, TC is a valid solution to develop hardware security capabilities in IoT devices interacting with the Cloud.

5 Registration Strategies of IoT Devices Joing the Cloud

The IoT device, e.g., the Arduino Yun extended with security capabilities, can follow two different registration methods:

– case A, Unsupervised: auto registration of MAC address and obH;
– case B, Supervised: end-user web registration of MAC address and obH.

In both the cases, the end-user needs to enable the IoT device to maintaing the WiFi network association using the wps button on his wireless AP. Hence, the IoT device can access the Internet performing the authentication as describe in Sect. 4. In the case B, the IoT device board flashes an orange LED, and after its partial registration it shows an orange fixed-on LED. The full registration is achieved when the end-user associates the IoT device board with his/her web profile. The users adopts a web site to register the board, in particular typing the MAC address shown in the external part of the box provided by the manufacturer. If the MAC in M match the MAC typed in the website, the board flashes a green LED, and the user can confirm the operation, otherwise (obtaining no flashing LED) he/she should repeat the procedure. After that, the full registration has been accomplished, the board shows a green fixed-on LED. Now the Cloud has the full control of the board, hence it can deploy firmware, managing configuration, install software and so on. The user only pushed a button (wps) and typed a code in the web site of the Cloud operator.

6 Conclusion and Future Work

In this paper, we discussed an approach to integrate the IoT with Cloud computing. In particular a system is presented analyzing the different elements involved and how they interact each other. Using the Arduino Yun as example, we discussed how IoT devices can be extended to support the interaction with the Cloud. In particular, we focused on a secure self-identification mechanism that allows a Cloud provider to deploy the firmware and configure the device. The paper deals with the design of secure IoT infrastructures. Currently, IoT devices are at the early stage and how argued in this paper, they are not ready yet to support complex Cloud scenarios, even though the roadmap toward innovative Cloud IoT services begins to be tracked. In future works, we plan to study the integration of the Trusted Computing in Arduino Yun for an advanced self-identification when the device joins a Cloud environment.

References

1. Unleashing potential of Future Internet and Cloud computing, November 2013. http://ec.europa.eu/digital-agenda/en/news/unleashing-potential-future-internet-and-cloud-computing
2. Trusted Computing Group (TCG). http://www.trustedcomputinggroup.org
3. Aslam, M., Rea, S., Pesch, D.: Service provisioning for the wsn cloud. In: 2012 IEEE 5th International Conference on Cloud Computing (CLOUD), pp. 962–969 (2012). doi:10.1109/CLOUD.2012.132
4. Ayadi, I., Noemie, S.: Adaptive provisioning of connectivity-as-a-service for mobile cloud computing. In: 2014 2nd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud), pp. 169–175 (2014). doi:10.1109/MobileCloud.2014.33
5. Celesti, A., Fazio, M., Villari, M.: Se clever: A secure message oriented middleware for cloud federation. In: IEEE Symposium on Computers and Communications (ISCC), pp. 35–40 (2013)
6. Chandra, A., Lee, Y., Kim, B.M., Maeng, S.Y., Park, S.H., Lee, S.R.: Review on sensor cloud and its integration with arduino based sensor network. In: 2013 International Conference on IT Convergence and Security (ICITCS), pp. 1–4 (2013). doi:10.1109/ICITCS.2013.6717876
7. Deshwal, A., Kohli, S., Chethan, K.: Information as a service based architectural solution for wsn. In: 2012 1st IEEE International Conference on Communications in China (ICCC), pp. 68–73 (2012). doi:10.1109/ICCChina.2012.6356972
8. Fazio, M., Paone, M., Puliafito, A., Villari, M.: Huge amount of heterogeneous sensed data needs the cloud. In: 2012 9th International Multi-Conference on Systems, Signals and Devices (SSD), pp. 1–6. IEEE (2012)
9. Suciu, G., Vulpe, A., Halunga, S., Fratu, O., Todoran, G., Suciu, V.: Smart cities built on resilient cloud computing and secure internet of things. In: 2013 19th International Conference on Control Systems and Computer Science (CSCS), pp. 513–518 (2013). doi:10.1109/CSCS.2013.58
10. Tu'n, A.L., Quoc, H., Serrano, M., Hauswirth, M., Soldatos, J., Papaioannou, T., Aberer, K.: Global sensor modeling and constrained application methods enabling cloud-based open space smart services. In: Ubiquitous Intelligence Computing, International Conference on Autonomic Trusted Computing (UIC/ATC), pp. 196–203 (2012)
11. Wang, Y., Lin, W., Zhang, T., Ma, Y.: Research on application and security protection of internet of things in smart grid. In: IET International Conference on Information Science and Control Engineering 2012 (ICISCE 2012), pp. 1–5 (2012). doi:10.1049/cp.2012.2311
12. Yao, X., Han, X., Du, X., Zhou, X.: A lightweight multicast authentication mechanism for small scale iot applications. IEEE Sens. J. **13**(10), 3693–3701 (2013). doi:10.1109/JSEN.2013.2266116