A framework for the secure storage of data generated in the IoT

Ricardo Costa and António Pinto

Abstract The Internet of Things can be seen has a growing number of *things* that inter-operate using an Internet-based infrastructure and that has evolved during the last years with little concern for the privacy of its users, especially regarding how the collected data is stored. Technological measures ensuring users privacy must be established. In this paper we will present a technological framework for the secure storage of data. *Things* can then interact with the framework's API much in the same way they now interact with its current servers, after which, the framework will perform the required operations in order to secure the data before storing it. The methods adopted for the secure storage will maintain the sharing ability, conveniently allowing authorized access to other users, the initial user's terms (e.g. data anonymity) and the ability to revoke assigned privileges at all times.

Key words: Internet, Things, IoT, Secure, Storage, Database, Framework, API

1 Introduction

In the last years we have witnesses a unparalleled growth in the use of devices by human beings, making the, so called, Internet of Things (IoT) the most hyped technology in the planet [1][2], and is expected to continue. Fig. 1 clearly shows that the IoT is the emerging technologies that is most expected in a 5 to 10 years space. Such tremendous, completely uncontrolled, growth in such a short time frame has had no support from the traditional IT Industry. The market urged to respond to the

Ricardo Costa
CIICESI, Escola Superior de Tecnologia e Gestão de Felgueiras, Instituto Politécnico do Porto, Felgueiras - Portugal e-mail: rcosta@estgf.ipp.pt

António Pinto
GCC, CIICESI, Escola Superior de Tecnologia e Gestão de Felgueiras, Instituto Politécnico do Porto, Portugal and INESC TEC, Porto, Portugal e-mail: apinto@inesctec.pt

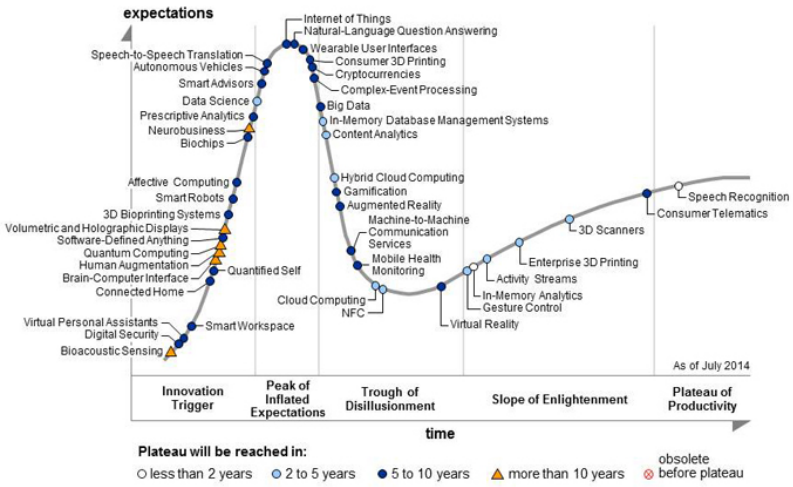

Fig. 1 Hype Cycle for Emerging Technologies, 2014

consumers needs (or need to consume) and that has raised some complex problems, being the security and privacy some of the major ones, in our opinion.

1.1 IoT by the Numbers

According to the ABI Research [3], the installed base of active wireless connected devices will exceed 16 billion in 2014, about 20% more than in 2013 and the number of devices will more than double from that to the 40.9 billion deviced expected for 2020. According to IDC, the IoT will represent a market value of $7.1 trillion in 2020, an impressive growth from the $1.9 trillion of 2013. Additionally, and just to name a few [4]:

- **IHS Automotive** says the worldwide number of cars connected to the Internet will grow to 152 million in 2020;
- **Navigant Research** says the worldwide installed base of smart meters will grow from 313 million in 2013 to nearly 1.1 billion in 2022;
- **On World** says the worldwide number of, Internet connected, wireless light bulbs and lamps will grow from 2.4 million in 2013 to over 100 million by 2020;
- **Juniper Research** says the wearables market will exceed the $1.5 billion in 2014, the double of its value in 2013.

Most of these devices, wearable or not, are trackers (or have the ability to track), or are health monitors, or record images and videos, among other functionalities. All of them, one way or another, collect data. This data is then sent to servers on the Internet that store it.

1.2 The Problem

The problem [5, 6], in our opinion, arises when data is put into the equation, specially if such data is considered sensible and private by its owner. Sensible and private data should not be stored in the traditionally way, due to its additional security and privacy requirements. This new kind of, massive collected, data is, however, being stored, without any special kind of security and privacy concerns, all around the world, in traditional databases (or new databases engines, but with the traditional store, search and access model) [6]. Due to the immense quantity of potential clients, effort has mainly been on server functionality, scalability, and responsiveness. Several solutions implement access control but store their data in clear text in their databases [7, 8, 9, 10, 11, 12]. This data is, in many cases, private data that can even be potentially used to harm the original data owner (the only real owner, in our opinion) without is expressed permission. Some example scenarios of such potentially harmful situations are describes next.

Scenario 1 - Health Data: Assume a scenario of a user that practices sports, like running or biking, and uses his smart watch/phone/band to monitor his heart rate. He does so for a long period of time, years maybe, and then has a heart related health problem. The stored data has the potential to be used to determine the risk of that person suffering another heart related health problem health and, ultimately, to be used by insurance companies to not insure, or to stop insuring that person.

Scenario 2 - Track Data: Assume a scenario of a user, a truck driver, that has agreed to use a, remotely accessible, GPS-enabled device so that its employer can better estimate the delivery times, and supply those times to the company costumers. The stored data has the potential to be used by the employer to monitor and control its employee even outside the normal, day-to-day, work hours.

Scenario 3 - Pictures: Assume a scenario of a user that subscribes, on its mobile smartphone, a picture backup service that automatically sends the pictures to a cloud-based storage. In order to maintain the backup service scalability and its server CPU usage within working limits, the service does not encrypt the pictures in its databases, requiring only some form of user authentication to allow access. Such service can potentially be exploited, by means of a 0-day bug, for instance, and the stored pictures can be widely divulged in the Internet exposing the private life of the picture owners.

Scenario 4 - Contact information: Assume a scenario of a user that uses a cloud-based contact information backup service where he stores all his mobile phone contacts (names, cell phone numbers, street addresses, email addresses). The stored data has the potential to be sold by the backup service provider to advert companies or to aggressive phone selling companies.

This scenarios present some of the potential uses that such kind of data can have, even without the knowledge of its owner. Additionally, there are several companies that have huge profits for using, free of charge, such private data without paying any kind of compensation to the owners of the data.

2 Background and related work

The work related to ours can by categorized into the following two approaches: 1) IoT access standardization, and 2) data acquisition.

The first approach focus on tackling the heterogeneity found on the diversity of equipment, communication protocols and communication technologies that make the IoT and does so by creating a framework that makes them accessible in a uniform manner. In [7], the proposed service oriented architecture that hides the uniforms access devices via web services. It also supports service management, device management and security. In [8], another framework that tries to standardize IoT device access is proposed, in this case by making use of RESTful web services and with their services implemented as a OSGi (Open Service Gateway initiative) specification. More recently, in [9], another RESTful-based middleware framework was also proposed, differing by its use of the MQTT protocol [13]. Another RESTful middleware was also presented in [10], this one focusing more on the interaction with devices by means of the OpenMTC framework.

The second approach focuses on data acquisition, integration and storage in a way that still enables access control and the management of security and privacy. In [11], a platform for data acquisition and integration is proposed for IoT. This platform, in a first phase, collects data from devices and stores it in a cloud environment and, in a second phase, context-oriented mechanisms are executed over the data to produce context data. Personal data vaults, proposed in [12], is referred as a privacy architecture where user retain ownership of their data. Data is said to be filtered before sharing and user can take part in controlled data-sharing decisions. Their work is based on fine-grained Access Control Lists (ACL) and the capability to trace and audit operations logs. Neither solution use data encryption has a way of guarantying privacy nor consider user monetizing strategies.

Name	Approach	Type	Access control	Data encryption	User control
SOCRADES	Access	Service-oriented	No	No	No
MAGIC Broker 2	Access	RESTful/OSGi	No	No	No
QEST	Access	RESTful/MQTT	No	No	No
M2M APIs	Access	RESTful	Yes	No	Unclear
Context	Data	OSGi	Yes	No	No
PDV	Data	Web-based	Yes	No	Yes

Table 1 Related work

Table 1 summarises the solutions that relate to our work. Each solution is identified by a name, is categorized by its approach and type, and analysed to verify if the solution implements access control mechanisms, if the data is encrypted prior to its storage, and if the user that owns the data is allowed to maintain control over the data sharing. For instance, the PDV (Personal data vault) solution falls in our data acquisition approach, is a web-based (HTTP) cloud-like solution that imple-

ments data access control, that allows the data owners to perform sharing control but stores its data in a clear text form.

3 The Secure Framework for IoT (Sec4IoT)

In the traditional IoT architecture (Fig. 2) the *things* interact with the servers via an Internet-based infrastructure. Normally, the only security-related aspects that are considered are the ones related to the communication channels, often recurring to SSL/TLS protocols as the foolproof solution. We believe this is not enough since all the data that is stored, is stored in clear text form in their databases. Additionally, except [12], no solution gives their users the capability to either control data sharing, to audit data sharing or to revoke data sharing.

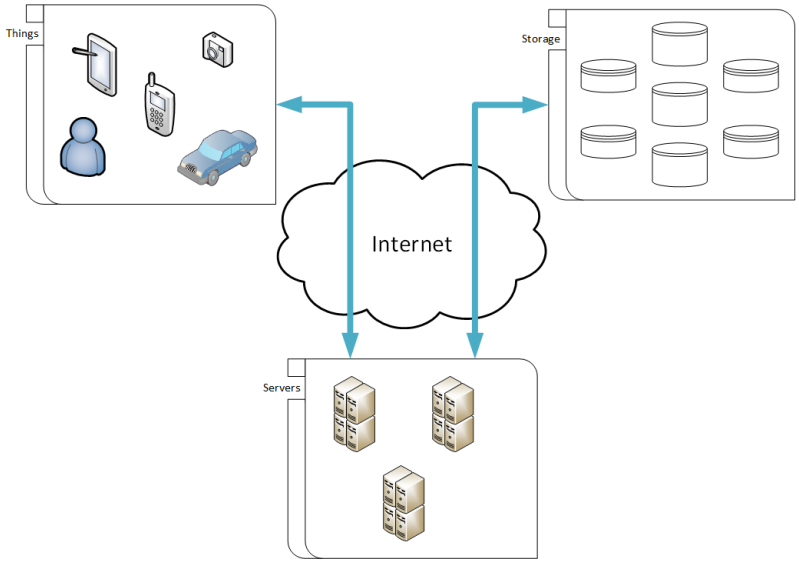

Fig. 2 IoT architecture

We propose adding an additional layer of security to the IoT tradition architecture (Fig. 3) and that such security layer be provided by our framework. Despite maintaining the traditional architecture and its communication channels, we require that the *things* use our framework API, named Secure Framework for IoT (Sec4IoT), for data sending and storing. The framework will implement an additional security layer providing, not only the needed user privacy, but also the user control of its own data.

This additional security layer will be explained in the next set of steps, using the notation presented in Table 2:

Notation	Meaning
pubK	Asymmetric cryptography public key
privK	Asymmetric cryptography private key
secK	Symmetric cryptography secret key
[item1, item 2]	Array containing item 1 and item 2
M_K	Message M encrypted with key K

Table 2 Adopted notation

1. **Initialization:** all the intervening parts of the process will need to pass by an initialization phase where each part will generate a public and private key pair [**?**];
2. The *thing* calls the API method for data upload - uploadData();
3. The uploadData() method will, before actually uploading any kind of data, generate a random secret key *secK*, encrypt the data with it and encrypt the *secK* with the user *pubK*, then it will generate a token *dataT* containing [$data_{secK}$, $secK_{pubK}$] and, finally, upload the token to the servers;
4. The server will receive the encrypted data token dataT and store it in the database or datastore.

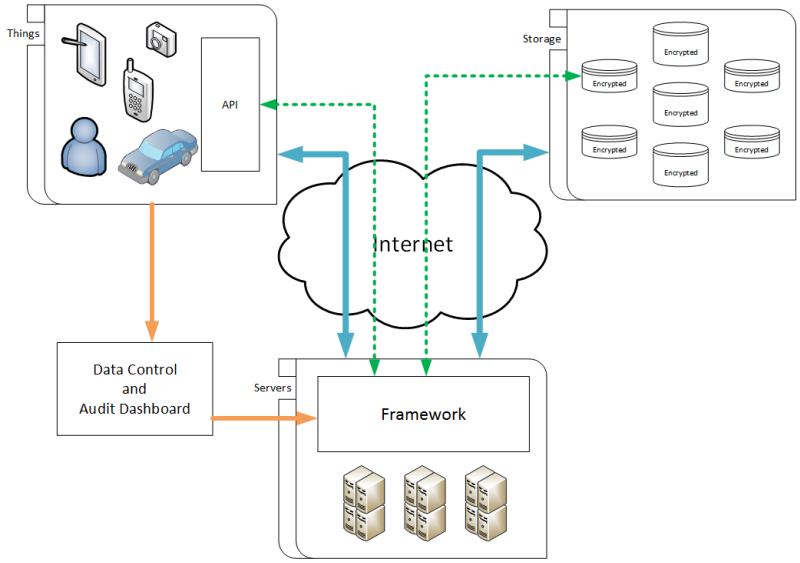

Fig. 3 IoT data storage new, framework, approach

This set of steps will guarantee that the stored data is only accessible by its true owner, the user. Additionally the framework API will give its users the possibility to share their data with other users or with a service provider, we expect that some services may require it, if he is willing to do so. To achieve that the only thing we

have to do is, after the user authorizes it, it to encrypt the user's data *secK* with the *pubK* of the provider or of another user, giving them the capability to decrypt the stored data. We will also enable a mechanism to anonymise data prior to its sharing.

The framework will contain a *Data Control and Audit* dashboard where the user will be able to know what accesses are being made to his data and by whom. He will also be able to revoke shares and, even, delete the data if possible (terms of service).

4 Conclusions and Future Work

We have presented a new framework that is able to provide the additional and highly needed layer of privacy and security to the traditional IoT architecture. This additional layer enables users to regain their privacy rights, to collect their data only for their own consumption or to share it with a service provider or with other users, all accordingly to his own rules. We believe that this framework can be seemly introduced into the already in production solutions, turning them more secure from that moment on. Additionally, the proposed framework relegates the privacy control to the users, relieving services of such a burden. We also believe that this framework can also be easily adopted by new or ongoing developments, making them easier to include the needed security, and thus, accelerating its development and deployment.

As future work, we plan to develop a proof-of-concept cloud-based data storage service that makes use of the proposed security framework and implements the mentioned data control and audit dashboard.

References

1. "Gartner's 2014 hype cycle for emerging technologies maps the journey to digital business."
2. "It's official: The internet of things takes over big data as the most hyped technology."
3. "The internet of things will drive wireless connected devices to 40.9 billion in 2020."
4. "Internet of things by the numbers: Market estimates and forecasts."
5. R. H. Weber, "Internet of things-new security and privacy challenges," *Computer Law & Security Review*, vol. 26, no. 1, pp. 23–30, 2010.
6. R. Roman, P. Najera, and J. Lopez, "Securing the internet of things," *Computer*, vol. 44, no. 9, pp. 51–58, 2011.
7. P. Spiess, S. Karnouskos, D. Guinard, D. Savio, O. Baecker, L. Souza, and V. Trifa, "SOA-based integration of the internet of things in enterprise services," in *Web Services, 2009. ICWS 2009. IEEE International Conference on*, pp. 968–975, IEEE, 2009.
8. M. Blackstock, N. Kaviani, R. Lea, and A. Friday, "MAGIC Broker 2: An open and extensible platform for the internet of things," in *Internet of Things (IOT), 2010*, pp. 1–8, IEEE, 2010.
9. M. Collina, G. E. Corazza, and A. Vanelli-Coralli, "Introducing the QEST broker: Scaling the IoT by bridging MQTT and REST," in *Personal Indoor and Mobile Radio Communications (PIMRC), 2012 IEEE 23rd International Symposium on*, pp. 36–41, IEEE, 2012.
10. A. Elmangoush, T. Magedanz, A. Blotny, and N. Blum, "Design of RESTful APIs for M2M services," in *Intelligence in Next Generation Networks (ICIN), 2012 16th International Conference on*, pp. 50–56, IEEE, 2012.

11. Y.-S. Chen and Y.-R. Chen, "Context-oriented data acquisition and integration platform for internet of things," in *Technologies and Applications of Artificial Intelligence (TAAI), 2012 Conference on*, pp. 103–108, IEEE, 2012.
12. M. Mun, S. Hao, N. Mishra, K. Shilton, J. Burke, D. Estrin, M. Hansen, and R. Govindan, "Personal data vaults: a locus of control for personal data streams," in *Proceedings of the 6th International COnference*, p. 17, ACM, 2010.
13. OASIS Standard, "MQTT Version 3.1.1," 2014.