User control over Cloud data security in an Internet of Things environment

Ștefan-Ciprian ARSENI
Faculty of Information Systems and Cyber Security, “Ferdinand I” Military Technical Academy, Bucharest
Faculty of Electronics, Telecommunications and Information Technology, University “Politehnica” Bucharest
stefan.arseni@mta.ro

Abstract: The proliferation of smart devices in our everyday activities has led to an increase in the adoption rate of the Internet of Things (IoT) concept. This evolution has also been enabled by other interesting technologies, such as Software-Defined Networks (SDNs) or Cloud computing. Besides the multiple benefits that the IoT paradigm brings, the large amount of private data accumulated by it raises new concerns about how this data is being processed and stored, especially in Cloud systems that are either abstract or untouchable for a regular user. The paper begins by briefly assessing different data storage solutions existing on the market and from the scientific community, ending with a proposal that was introduced last year, of an adaptive storage solution that enables end-users to control the security mechanisms used in the their data storing process.
Keywords: Cloud storage, data security, user-controlled cryptographic methods, Internet of Things.

INTRODUCTION

One of the main challenges in using information technologies available today is the selection of the “best” one that empowers us to solve our problems in the shortest period of time. Yet, in many cases, this improvement of services is mainly the result of integration between multiple technologies, each one requiring different types of data to properly execute their tasks. The complexity of architectures, corroborated with this increase in data demand, led to the appearance of security concerns among end-users.

Internet of Things (IoT) is one of the main examples that fits in this category of complex systems integrated in many layers of our society. Enabled by other new technologies, such as Software-Defined Networks, Cloud computing, or the introduction of 5G communication devices, IoT becomes the main network for collecting user or environmental data. If, in the case of Smart-City or Smart-Logistics scenarios, collected data can be considered to be public, thus requiring a lower minimum level of protection, in the case of other scenarios, such as Smart-Home or Smart-Health, collected data is more personal for an user, thus it needs to be properly secured.

The need of security in IoT environments is augmented also by the nature of the underlying IoT devices: interconnected embedded elements with a reduced set of computational resources (e.g. battery, processing performance or storage capacity). Even though the device layer of an IoT system has a wider range of vulnerabilities, the remaining layers are also vulnerable to certain types of attacks, as stated by (Sicari et al., 2015) and (Alaba et al, 2017) , and are more targeted by malicious users, given that these are the layers that aggregate, process and store data captured from the underlying devices. Therefore, the need for secure processing and storage of sensible data is oriented mainly at the Cloud layer, where most of the persistent data is being stored. Still, besides the IoT context, secure data storage on local or remote systems is a subject of interest for other domains, expanding from global companies to local businesses and, even, individuals demanding easy access to their data from different locations.

Trying to adapt and make a compromise between security and usability, the information security products and systems available today have assumed the aspects of how and what security mechanisms are applied to data, freeing customers of these types of decisions. Even though this simplifies the procedures a user needs to pass through before using a storing service, it also means that the primary element of a security chain, the data encryption keys, are out of the reach of the user, being created and managed only by the storage service provider.

CLOUD STORAGE - A SECURITY OVERVIEW

In Cloud computing technologies, security concerns can be divided into provider-related and user-related vulnerabilities. In this paradigm, the provider will ensure the security of his facility and services, offering guarantees to end-users that their data is protected against unauthorized access or alteration. (Chen & Zhao, 2012) and (Albugmi et al., 2016) have made an analysis of data security and privacy protection issues associated with Cloud computing services and present some of the solutions available on the market. They conclude by emphasizing the need for integrating the data security aspects in the early stages of designing Cloud-based applications.

Taking into considerations the complex process of implementing secure local storage facilities, many companies have opted, depending on the type of data they process, for either a Hybrid or a Public Cloud service. There are currently many providers of such services, the most well-known being Amazon (AWS), Microsoft (Azure), Google or IBM.

Unlike organizations that have the means to purchase complex and expensive Cloud platform services, individual users can benefit from the data storage solutions provided by companies like DropBox, Google (Google Drive) or Microsoft (OneDrive). This type of services, known as Storage-as-a-Service (SaaS) (Kundu et al, 2010), offers some advantages for end-users, including: access from anywhere, improved communication, minimal administration and, implicitly, reduced infrastructure costs for end-users. SaaS can be divided into three main models:

- ***Storage without encryption*** – in this kind of storage, data is stored as plaintext, without applying any encryption algorithms on it. Still, some security mechanisms must be implemented, to ensure, at a minimum, the integrity of that data;
- ***Storage with server-side encryption*** – in this case, data is stored in an encrypted form on the server side. The encryption is made after the Cloud storage solution receives the data, but before it is written to disks. The cryptographic keys used for data encryption are generated, stored and used by the server. For this type of storage, the user cannot interfere with the cryptographic keys' lifecycle and the cryptographic mechanisms used, thus a certain degree of trust must exist between the user and the provider. Data is automatically decrypted, on the server side, when accessed by an authorized user.
- ***Storage with client-side encryption*** – in this kind of storage, data is stored on the server side in an encrypted manner, but the encryption is made before data is sent to the Cloud storage provider. The cryptographic keys used for data encryption are generated, stored and managed on the client-side application. This implies the Cloud has no knowledge of the cryptographic mechanisms and keys used for encryption and data storage. A corrupt or malicious Cloud service provider could still modify the transmitted data, by altering it at any moment starting from receiving the encrypted data to when a user downloads it. This issue can be easily mitigated by using cryptographic mechanisms that ensure not only the confidentiality, but also the integrity of the encrypted data.

Researchers used different methods or solutions to address the issues identified in most of the Cloud storage systems available today. One of the methods presented (Zhou et al., 2015) is to implement a hybrid encryption and access control scheme that will ensure a role-based access control. In another paper, (Hwang et al., 2011) present a generic business model in which the encryption and decryption services can be separated from the storage service. In recent years, research in this field has intensified and each one of the newly proposed solutions is designed to improve the security level ensured by a Cloud storage system but the need to trust the service provider has not been eliminated.

CLOUD STORAGE FROM THE IoT PERSPECTIVE

In the IoT paradigm, Cloud platforms are often viewed as a middleware composed of a multitude of technologies that enable end-users to extract and analyse data from different IoT devices. Taking advantage of the improvements of Cloud computing, IoT Cloud platforms became modules overseeing entire IoT systems, being able to execute complex data processing requests and manage underlying IoT subsystems placed in multiple locations. Despite this evolution, in recent years, IoT systems started to move a part of their Cloud services in the lower layers of the IoT environment, in the Edge or Gateway layers. Still, the storage component remained in the Cloud, given its central position in an IoT system and the possibility to reach it from any location.

When analysing the security mechanisms implemented at the Cloud layer of an IoT system, the generic Cloud services are activate and establish a well-defined security perimeter. Yet, some differences appear when taking into consideration the nature of the IoT devices that will be connected to the Cloud platform and what services are going to be used by these devices. These elements will be emphasized in the following paragraphs of this section, containing brief descriptions of some well-known Cloud platforms for IoT that can be found on the market.

Amazon AWS IoT

The Amazon AWS Cloud platform for IoT (Amazon AWS IoT) is a complete Cloud solution for an IoT system, providing a wide range of services, from authentication to analysis and report services (AWS IoT, n.d.). As a means to connect to this platform, Amazon AWS IoT requires the existence of digital certificates or other security objects that contain valid credentials. After connecting, the IoT device can access shared resources through different services. How each device accesses these resources and how messages uploaded to the Cloud platform are filtered are the tasks of the IoT system administrator. Even though a certain degree of control over the security mechanisms is directed towards the end-user, the main elements of the secure data storing process are not reachable, being controlled internally to the Cloud solution.

Microsoft Azure IoT Suite

Through Azure IoT Suite (Azure IoT, n.d.), Microsoft proposes a solution that addresses the entire value chain of an IoT system, offering modules for each layer (device, gateway and Cloud). After successfully registering a device on the Cloud platform and retrieving the required credentials, an administrator can create different access control policies that will be enforced when devices will connect to the Cloud platform services. As a differentiating element, the Cloud module allows the administrator to customize which security mechanisms are used for securing the data. Still, the encryption keys are generated and managed by Microsoft for the entire lifecycle of the data.

Google Cloud IoT Core

Similar to the two previously presented platforms, the Google IoT Core Cloud platform (Cloud IoT Core, n.d.) offers multiple services to enable the connection to an IoT system. While the means of authenticating devices and securing communication links are secured with security objects that an administrator can manage and customize, the process of secure data storage is still out of reach for the administrator.

OpenIoT

Designed and developed as a research project funded by the European Union, OpenIoT (OpenIoT, n.d.) is a set of software modules that can be used as extensions for existing Cloud platforms, making them adaptable to an IoT environment. Given that OpenIoT does not require the creation of a new Cloud platform, the majority of existing security services will be reused by it. An exception can be seen in the method of authenticating devices and enforcing resource access lists over shared resources, mainly the Cloud storage. A new Central Authentication Services (CAS) will create a security token that will enable a user or devices to access different resources, for a specific period of time. After expiration, this token will be renewed, following the same procedure. Neither the user nor the device have any control over what methods are run to create this token or how it is managed by the Cloud platform.

CLOUD STORAGE WITH CLIENT-SIDE ENCRYPTION

By design, the client-side encryption storage enables a zero-knowledge Cloud environment, thus end-users can have complete trust that their data will not be divulged in case of a breach in the Cloud storage provider security perimeter. But, as a downside to high security, end-users are now the sole entity that has control over the encryption keys and cryptographic algorithms. Thus, they need to enforce security mechanism on their local machine and, preferably, use special equipment that can provide the required protection for their encryption keys, such as a security token or smart card. In this endeavour to create Cloud storage with client-side encryption, several companies proposed different solutions that can be seen on the market, but the scientific community has also been active in creating new models or specifications for how this concept can be integrated in the generic Cloud computing paradigm.

SpiderOak (SpiderOak Secure Software, n.d.) and Tresorit (End-to-End Encrypted File Sync&Sharing, n.d.) are two of the most used commercial solutions that offer this type of Cloud storage. Both solutions have architectures created based on the “Zero-knowledge” concept, that means they only store encrypted data, without having any persistent data related to what password or keys have been used by the client to secure the uploaded data. Passwords are stored on the client machine and encryption keys are created either using Password-Based Key Derivation Functions (PBKDFs) or random generation.

With TwinCloud, (Bicakci et al., 2016) propose a different approach by using two or more Cloud providers for ensuring a high level of security. The increased number of Cloud providers is due to a split in the key management service that the authors propose. This split results in encrypted data being stored on one Cloud, while the encryption keys are being sent to another Cloud provider. By doing this, authors state that the problems of complex key management that arise when sharing a file between users can be mitigated. This architecture is based on the supposition that two Cloud providers will not divulge information between themselves and if one of these two Cloud platforms is breached, data will not be revealed.

From an IoT perspective, the client-side encryption can prove to be either a simple and useful integration or a complex process that will be either impossible to integrate, given the limited resources available on the terminal devices, or difficult to manage, in case of a collaborative IoT system, where data needs to be shared between multiple groups and devices. Still, if the client-side encryption approach can provide security benefits, even the second scenario can be implemented, by moving the logic of client-side encryption from the terminal devices to the gateway, a device that is under the complete control of the administrator.

AN ADAPTIVE CLOUD STORAGE SOLUTION WITH CLIENT-SIDE ENCRYPTION

The issue of complete control not only over data collected from an IoT system, but also over the encryption mechanisms used to secure that data, has been a matter of interest in the research community in recent years, fuelled mainly by the tendency of organizations to collect more and more personal data from users, in order to provide services customized for each individual user.

As presented in the previous sections, the inability to control encryption-related elements in existing Cloud platforms led to the growth of organizations offering client-side encryption solutions that can address this matter. Still, even this solution let administrators create and manage only the encryption keys, while cryptographic algorithms remain unchanged. This can, sometimes, prove to be insufficient, mainly when requiring the usage of complex cryptographic schemes on devices that have limited resources. To address this problem, (Arseni et al., 2018) presents a proposal for a customizable Cloud storage solution with client-side encryption. The paper only introduces the solution in the context of large data processing. Taking into consideration the IoT context, the proposed solution can easily be adapted to this new working environment, while maintaining its desired advantages. Fig.1 presents an overview of the iterated solution in the IoT context, emphasizing the involved entities and security elements.

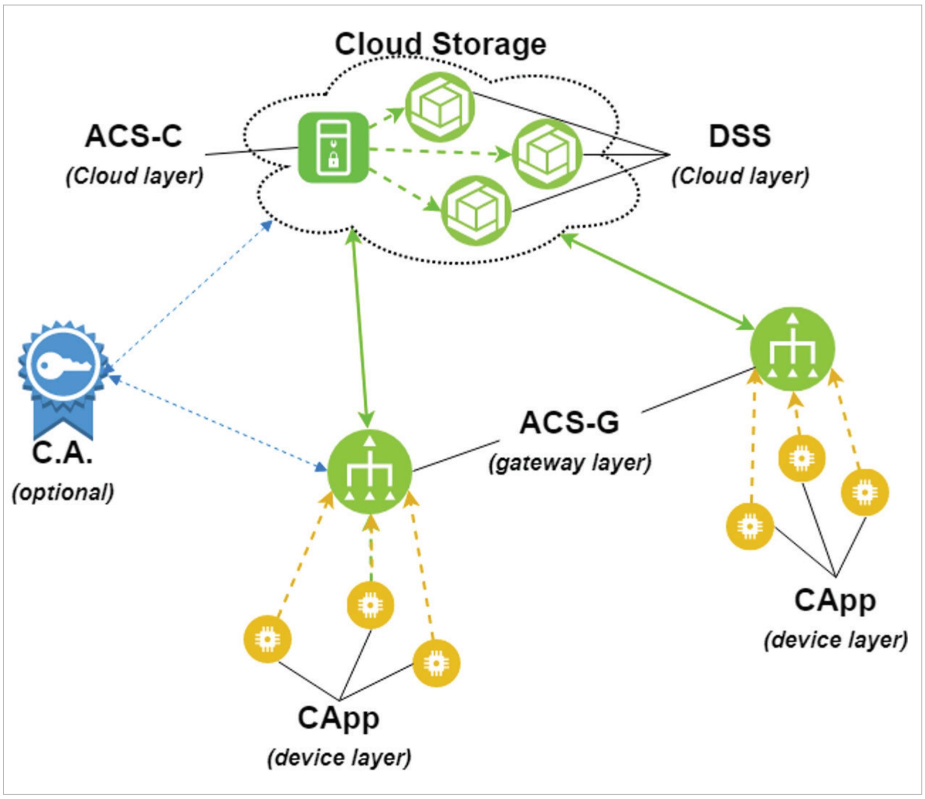

***Fig. 1:** The elements of the proposed solution (Arseni et al., 2018)*

This new iteration of the solution is built upon the same functional methods that the first proposal has, namely:

• a Client Application (CApp) that contains the logic of generating symmetric cryptographic keys, based on the NIST SP 800-90A standard (Barker & Kelsey, 2015), and encrypting data using a custom cryptographic algorithm installed by the user;

• a Data Storage Server (DSS) that represents the Cloud storage provider where encrypted data will be stored;

• an Access Control Server (ACS) that is now divided into two components: ACS-C placed on the Cloud component (for user management) and ACS-G placed on the IoT gateway (for device management). It is responsible for:

- user authentication and authorization based on valid credentials (Cloud);
- enforcing access control lists for authorized users and devices (Cloud);
- authenticating devices and validating their request to push data to DSS (gateway);
- acting as a relay whenever a file sharing demand is received from a valid user (gateway).

The fourth component in the first proposal, the Certificate Authority (C.A), maintains its third party status and can be considered optional in this new solution, given that the device certificates can be also created by the administrator, because the entire flow of data from devices to the Cloud passes through the gateway. Still, the C.A. can be used for validating certificates by the ASC-C (certificates of users and gateways that connect to DSS) or by the ASC-G (certificates used by the Cloud to secure the transmission channel).

As an adaptation to the IoT context, the CApp component contains the minimum required set of instructions and allows the integration of lightweight cryptographic algorithms as a mechanism for encrypting data. Thus, the requirement of having resource consuming cryptography on IoT devices was eliminated without affecting the security level of the overall system.

CONCLUSIONS

Among other new technologies that bring improvements to our daily task resolution, Cloud computing is a key enabler for another new paradigm, Internet of Things. Even if the security mechanisms that ensure data protection and authorized access to it in the Cloud platform have evolved and maintain an adequate level of protection, recent breaches have left the general public with a feeling of insecurity. The Cambridge Analytica data scandal showed that there are still faulty mechanisms in place, and that, once data is uploaded onto a server, a user can only trust that it is being protected and not used for other purposes. In this new context of society adapting to intelligent environments enabled by IoT, data security is becoming a more pressing issue, and Cloud storage is the main actor.

As a response to this changing technological environment, well-known organizations have adapted their Cloud solutions to respond to specific security issues that arise in the IoT context. Still, the matter of user control over the security mechanisms for data protection has only been partially addressed, as was presented in the previous sections. Client-side encryption tries to empower users by having complete control over their data, using Cloud storage services only as a repository with access control and authentication, while the encryption of data and encryption keys management logic is moved onto the client application. Depending on the environment in which this methodology will be implemented, it can either be an easy or a challenging task that will require more derivation of existing software modules.

Several efforts in this area of client-side encryption have been presented as proposals, while others have been implemented and are available for use. Trying to cover both the classic user – Cloud and IoT device – Cloud scenarios, this paper proposes a reiteration of the solution for Cloud storage using client-side encryption, which was proposed by (Arseni et al., 2018). The iteration contains minor changes to the former architecture, involving only a shift in how some services are placed in different layers of the IoT system.

Given the higher level of technological competency that users need to attain before adopting a client-side encryption solution for their IoT systems, this concept has a slower adoption rate. But this is already starting to change, given that the rapid adoption of IoT will require users to show more interest in how their data is being manipulated and stored.

REFERENCE LIST

Alaba, F. A., Othman, M., Hashem, I. A. T., & Alotaibi, F. (2017). Internet of Things security: A survey. Journal of Network and Computer Applications, 88, 10–28. doi: 10.1016/j.jnca.2017.04.002.

Albugmi, A., Alassafi, M., Walters, R., & Wills, G. (2016). Data security in cloud computing. 5th International Conference on Future Generation Communication Technologies, 55–59. doi: 10.1109/FGCT.2016.7605062.

Arseni, S.-C., Rădoi, I., Măluțan, S.-B., Lazăr, M., & Dragomir, I.-R. (2018). A Data Storage Model with User Controlled Cryptographic Mechanisms for Data Processing. 2018 International Conference on Communications, 523–528. doi: 10.1109/iccomm.2018.8484804

AWS IoT: platforms, connectivity, applications and services. (n.d.). Retrieved September 27, 2019, from https://aws.amazon.com/iot/.

Azure IoT. (n.d.). Retrieved September 27, 2019, from https://azure.microsoft.com/en-US/overview/iot/.

Barker, E., & Kelsey, J. (2015). Recommendation for random number generation using deterministic random bit generators. Gaithersburg, MD: U.S. Dept. of Commerce, National Institute of Standards and Technology. doi: 10.6028/NIST.SP.800-90Ar1.

Bicakci, K., Yavuz, D. D., & Gurkan, S. (2016). TwinCloud: Secure cloud sharing without explicit key management. 2016 IEEE Conference on Communications and Network Security, 581–585. doi: 10.1109/cns.2016.7860552.

Chen, D., & Zhao, H. (2012). Data Security and Privacy Protection Issues in Cloud Computing. 2012 International Conference on Computer Science and Electronics Engineering, 647–651. doi: 10.1109/iccsee.2012.193.

Cloud IoT Core | Google Cloud. (n.d.). Retrieved September 28, 2019, from https://cloud.google.com/iot-core/.

End-to-End Encrypted File Sync & Sharing. (n.d.). Retrieved September 28, 2019, from https://tresorit.com/.

Hwang, J.-J., Chuang, H.-K., Hsu, Y.-C., & Wu, C.-H. (2011). A Business Model for Cloud Computing Based on a Separate Encryption and Decryption Service. 2011 International Conference on Information Science and Applications, 1–7. doi: 10.1109/icisa.2011.5772349.

Kundu, A., Banerjee, A., & Saha, P. (2010). Introducing New Services in Cloud Computing Environment. International Journal of Digital Content Technology and Its Applications, 4, 143–152. doi: 10.4156/jdcta.vol4.issue5.17.

OpenIoT – Open Source cloud solution for the Internet of Things. (n.d.). Retrieved September 28, 2019, from http://www.openiot.eu/.

Sicari, S., Rizzardi, A., Grieco, L., & Coen-Porisini, A. (2015). Security, privacy and trust in Internet of Things: The road ahead. Computer Networks, 76, 146–164. doi: 10.1016/j.comnet.2014.11.008.

SpiderOak Secure Software. (n.d.). Retrieved September 28, 2019, from https://spideroak.com/.

Zhou, L., Varadharajan, V., & Hitchens, M. (2015). Trust Enhanced Cryptographic Role-Based Access Control for Secure Cloud Data Storage. IEEE Transactions on Information Forensics and Security, 10(11), 2381–2395. doi: 10.1109/tifs.2015.2455952.