An adaptive authentication and authorization scheme for IoT's gateways: a blockchain based approach

Achraf Fayad, Badis Hammi, Rida Khatoun

An adaptive authentication and authorization scheme for IoT's gateways: a blockchain based approach

Achraf Fayad, Badis Hammi, Rida Khatoun
Telecom Paristech, France
achraf.fayad, badis.hammi, rida.khatoun,@telecom-paristech.fr,

***Abstract*—Security of the Internet of Things represents a field that strongly attracts academia and industry since it represents one of the main obstacles in its adoption. In this area, authentication and authorization methods holds a golden place in priority rank. Indeed, current approaches suffers from numerous limits. Moreover, generally, deployment systems use separately two methods one dedicated to the authentication and the other to the authorization, while the number of methods that combine both requirements is limited. In this work we propose an adaptive blockchain based authentication and authorization approach for IoT use cases. We provided a real implementation of our approach using *Java* language. The extensive evaluation provided, shows clearly the ability of our scheme in meeting the different requirements, as well as its ability in ensuring a very lightweight cost.**

***Index Terms*—Authentication, Authorization, Internet of things IoT, Privacy, Smart City, Smart home**

I. Introduction

The Internet of Thing (IoT) represents currently a part of our everyday lives and habits. This makes its evolution in exponential growth. Indeed, the number of connected devices such as surveillance cameras, refrigerators, electronic locks, connected cards, smart TVs, etc. is exploding. According to a recent *Gartner* study, 50 billion connected devices[1] will be deployed by 2020 [1]. IoT has the potential to introduces and develop a smart world by the development of new applications in different domains like smart homes, smart health, smart cities, industry 4.0, Wireless Sensor Networks (WSN), smart agriculture, etc. [2].

In IoT, each physical or virtual device should be reachable and produce content that can be retrieved by users regardless of their location. However, It is very important that only authenticated and authorized users make use of the system. Otherwise, it will be prone to numerous security risks such as information theft, data alteration and identity usurpation. Indeed, security issues remain the major obstacle to the large scale adoption and deployment of IoT since it is highly vulnerable to attacks for multiple reasons such as: (1) most of the communications are wireless, which makes the system more vulnerable to numerous attacks such as identity spoofing, messages eavesdropping, messages tampering and other security issues, and (2) multiple types of devices have limited resources in terms of energy, memory and processing capacity, which prevent them from implementing advanced security solutions [3].

IoT is not a simple technology, but a combination of multiple technologies achieving multiple use case scenarios. In many research works IoT is considered as a system-of-systems [3][4][5]. In many IoT systems and use cases, the used architecture relies on a gateway to ensure the majority of its activity. For example, the Figure 1 describes three use cases: (1) in a WSN, all the sensors upload some requested data (e.g. temperature, water level, etc.) to a gateway (also called sink). (2) in a Wireless Body Area Network (WBAN) all the monitoring sensors and motion detectors send and upload their data to a gateway (also called, sink or personal server). (3) In a smart home scenario, all the house components interacts with a gateway which also manages the users remote access. It exists numerous other IoT use cases where a gateway is the heart of the architecture, thus, the authentication and authorization applications are performed at the gateway level [6] [7].

Current use of gateways represents some limits:

1 - The non heterogeneity of authentication approaches: generally, all the nodes related to a gateway use the same authentication method. For example in WSNs a Pre-Shared Key (PSK) is used to authenticate the sensors. In other use cases, if the gateway use certificate based authentication, all the nodes must implement this same method. However, in numerous cases, the network is composed of different types of devices, each with its computation and energy capacities. For example in a smart home, if some objects, such as heat sensors, can only use PSK based authentication, since symmetric cryptography is less costly than other methods [8], other more powerful objects, such as smart fridges and TVs can easily use public key cryptography based methods, nonetheless, they still use PSK, since the gateway impose it to satisfy all the existing objects' types. **Thus, there is a need to propose an authentication mechanism that adapts to heterogeneous authentication techniques in order to ensure flexible and more resilient security.**

2 - The non mobility of nodes: generally, each device is associated to one gateway. However, in some use cases, some nodes can have mobility features (a sensor which location is changed or any other mobility case). To the best of our knowledge, nodes that have mobility features from a gateway to another are not authenticated at this gateway level but using a central server/service accessible through Internet (e.g., Intelligent Transportation Systems). Nevertheless, realizing

[1]In the remaining of this paper, we use indifferently the terms device, thing, object and smart thing in order to refer to a connected smart thing.

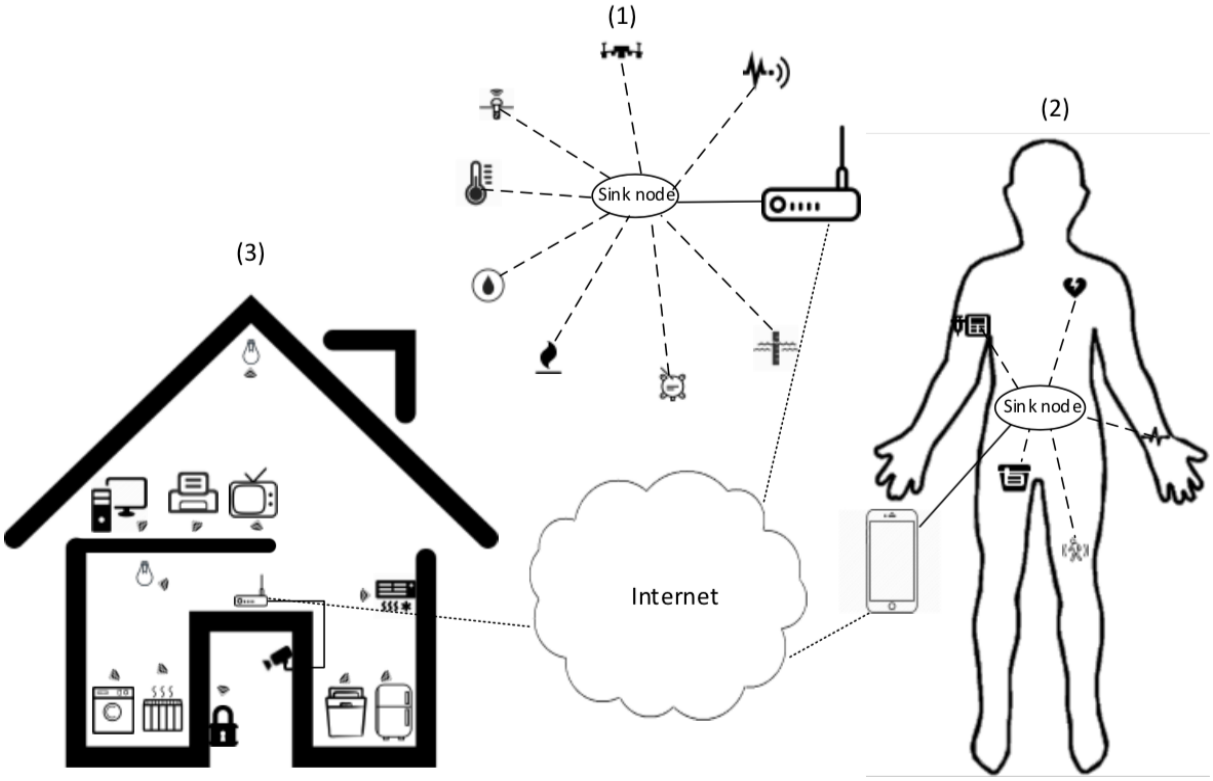


Fig. 1: IoT use cases

authentication at the gateway level can be less time consuming. Moreover, relying on centralized services in an environment such as IoT where the number of devices is exploding (countable in billions) can represent a real bottleneck. **Consequently, there is a need to propose a scalable security solution that allows the mobility of nodes while ensuring their authentication at the gateway level.**

3- Initialization phase: in the majority of systems, if a new device is added, a physical intervention on the gateway is needed to set up and configure the credential of this new device (e.g., adding the PSK of a new sensor on the gateway). However, knowing that the number of objects is exploding, this feature represent a real brake. **Hence, there is a need to propose a security solution that allows an easy integration of new devices as well as new services.**

Contribution

We believe, as many researchers [3][9][10][11] that blockchains represent a very promising technology for the development of decentralized and resilient security solutions in IoT context. Therefore, in this paper we propose a blockchain based authentication and authorization approach for IoT use cases. In our decentralized approach, (1) any device can be added without any physical intervention, (2) can move freely and still be authenticated and authorized at the gateway level and (3) it meets the scalability requirements of IoT.

This paper is organized as follows. Section II describes the related works. Then, Section III exhibits the architecture and the details of our approach. Finally Section V concludes the paper and introduce our future work.

II. Related works

There have been numerous works that tackled authentication and authorization in IoT [12][13][14]. Thus, in this section we describe the main known approaches.

Symmetric-key based authentication schemes are widespread since they are more suitable for constrained devices [15][16][17]. Although this solution is lightweight and fast, it does not meet all the performances requirements such as scalability and mobility of nodes.

To meet these requirements, numerous other works adopted public key cryptography [18][19] and even used X.509 [20] standard to ensure authentication. Indeed, Transport Layer Security (TLS) [21] has been recommended by many standards specified by Internet Engineering Task Force (IETF) for security services [14]. However, it is proven that TLS is not a wise choice with respect to the security best practices in IoT. In fact, TLS runs normally in a reliable transport protocol like TCP which is not always suitable for constrained resource devices, due to its congestion control algorithm [14]. As a replacement for TLS in the tightly constrained environments, the Datagram Transport Layer Security (DTLS) [22] protocol has been proposed recently. It operates over the unreliable transport protocol i.e., UDP and provides the same high security levels as TLS. However, it still rely on public key cryptography, which suffers from the high computational cost

and energy consumption and thus makes it not really suitable for numerous types of smart things.

In the last years, many works aimed the proposal of IoT security systems that rely on blockchains. Indeed, blockchain usage provides a decentralized architecture, anonymity of users if wished, tamperproof record of data transactions and highly trusted data verification [23].

Dorri et al. [23][24] propose a blockchain based architecture for IoT. Their approach relies on three interconnected blockchains: a local blockchain (private) for each use case, a shared blockchain (private) and an overly blockchain (public). Even if the solution resolve the problem of identification, it has multiple shortcomings like (1) each operation engender at least 8 network communications which can flood quickly the whole communication medium in case of high activity of nodes; and (2) the local blockchains are not distributed but centralized which is contrary to its principle because it can limit its power and availability.

Hardjono et al. [25] propose *ChainAnchor*, a privacy-preserving method for commissioning an IoT device into a cloud ecosystem. ChainAnchor supports device-owners being remunerated for selling their device sensor-data to service providers, and allow device-owners and service providers to share sensor-data in a privacy-preserving manner. However, Its goal is the full anonymity of the participating devices and is not adapted to numerous IoT use cases where the identification is needed.

In [10] the authors propose a robust, lightweight and energy-efficient security protocol for the WSN systems that rely on blockchains. However, this work was proposed for OCARI [26], a very specific WSN architecture.

In a previous work [3] we proposed Bubbles of Trust, an efficient decentralized authentication mechanism. This mechanism was implemented upon the public blockchain *Ethereum*, and aims at the creation of secured virtual zones, where devices can communicate securely.

To summarize, the majority of the existing approaches does not meet all the performances and security requirements such as the mobility of nodes and the adaptability to numerous security mechanisms.

III. Towards an adaptive authentication and authorization approach

Our approach can be applied to a huge number of IoT use cases and does not require special hardware. In this section we describe the different details related to the design and the functioning of the proposed approach. More specifically, our approach relies mainly on a the usage of a blockchain. A blockchain is defined as a distributed database (ledger) that maintains a permanent and tamper-proof record of transactional data. A blockchain is completely decentralized by relying on a peer-to-peer network. More precisely, each node of the network maintains a copy of the ledger to prevent a single point of failure. All copies are updated and validated simultaneously [10][27].

Currently numerous works explore blockchain applications in multiple use cases in order to develop secure decentralized applications [23][28]. Some Blockchains like *Ethereum* and *Hyper Ledger* facilitate the implementation of decentralized applications without the need for the modification of the blockchain's main source code, via what is commonly called *Smart Contracts* (called *Chaincodes* in *Hyper Ledger*).

A. Threat model

In this section we present our threat model. The latter is similar to the *Dolev-Yao* model [29].

1) Network model: The overall purpose of an authentication scheme is to allow multiple nodes to communicate in a trustworthy way over a non trusted network. In this work we consider a network that owns a set of devices offering and using different IoT services in a centralized or a distributed architecture. Each device communicates with a large number of other devices, however, each device depends on one or more gateways. Each gateway must hold a copy of the used blockchain. Exchanged messages pass through an unreliable and potentially lossy communication network, such as Internet. We also assume that all participants cannot be trusted. Indeed, the high number of smart things in the network, increases the risk of including compromised ones. Furthermore, the existing devices are of heterogeneous types and do not belong to the same use case. The network function consists in only forwarding packets and does not provide any security guarantee such as integrity or authentication. Thus, a malicious user can read, modify, drop or inject network messages.

2) Attacker Model: In this work, we assume that an attacker or malicious user has a total control over the used network i.e he can selectively sniff, drop, replay, reorder, inject, delay, and modify messages arbitrarily with negligible delay. However, the devices can receive unaltered messages. Nonetheless, no assumptions on the rate of the altered messages are made. Besides, the attacker can benefit from a computation power and storage larger than the implemented devices.

B. Initialization phase

Our approach relies mainly on the usage of a blockchain. Thus, when a new device is added to the network, the user that adds it must register it. This is provided by sending a transaction to the blockchain, which contains the following information about the added object: (1) the object's ID; (2) the authentication method, e.g., PSK, One Time Password (OTP), Certificate, etc.; (3) the authentication parameters e.g, the PSK, the object's public key, the object's certificate, etc.; (4) the authorizations list e.g., only upload to server IP on $Port$, upload/download to/from All, upload data of maximum 10 bytes per session, etc. Except the object ID, all the parameters are encrypted as shown in the example of the block content bellow.

```
=======================================
ObjectID: E43AC16A93
=======BEGIN ENCRYPTION=============
Authentication Method: PSK
Authentication Params: DE6S4ZB$CN1U0
Authorization List: {
Only Download From IP:Port
Only Upload To     IP:Port
}
=======END ENCRYPTION==============
=======================================
```

In this paper, we focus mainly on the protocol exchanges between the objects and the gateways. The details regarding (1) how the informations are encrypted in the Blockchain and which encryption type is used, (2) the blockchain's type and choice, (3) how the gateways downloads safely the blocks and decipher them, is the subject of a future work.

C. System's functioning

First, when a device wants to establish a communication session with the gateway, it sends it a Session Establishment Request (*SEReq*) that contains the object's ID and its authentication parameters (e.g., PSK). When the gateway receives the request, relying on the Object ID, it downloads, from the blockchain, the block containing the parameters related to that requester object. Then, the gateway decrypts the block and according to the retrieved parameters, it triggers the authentication operation. Afterwards, a Session Establishment Response (*SERep*) is sent to the device to inform it whether it is successfully authenticated or not. Finally, if there is a successful authentication, then, the session establishment can be set up.

Once the device is successfully authenticated and the session is established, the gateway controls each exchange and communication of the object relying on the list of authorization downloaded within the block. Indeed, since the gateway represents the sole link between the object and all its surroundings, it can control each action outgoing/coming from/to the object.

In our approach, any existing authentication method can be used as it is designed. Thus, the *SEReq* parameters are secured as it is the case in the method's classical use. Nonetheless, If the authentication method requires the usage of secret information like PSK, as highlighted by Figure 2 we propose a customized *SEReq* format to protect against man in the middle combined to cryptanalysis attacks as bellow:

```
====================================
\textit{SEReq} = [ID, HMAC{PSK || Nonce
       || Authentication Method}, Nonce]
====================================
```

Once the gateway receives this type of request, it retrieves the needed authentication parameters from the blockchain according to the received object ID. Then, it computes the HMAC{PSK ‖ Nonce ‖ Authentication Method} using the received Nonce and the deciphered data. If the obtained HMAC[2] matches the received one, it successfully authenticates the requester device.

IV. Evaluation an Discussion

A. Context and use case scenarios

As described earlier, the main advantage of our proposed approach relies in its suitability to the majority of IoT scenarios, all within ensuring an easy integration of new devices and services. In this section we evaluate our approach regarding its execution time as well as its financial cost. For this evaluation we choose two different use cases of the authentication between a gateway and an object:

1) an authentication via a PSK;
2) an mutual authentication using certificates.

Knowing that the usage of the blockchain induces an additional cost, for both scenarios, we measure the time needed for the authentication: (1) in the classical scenario, without relying on our approach; and (2) using our adaptive approach. Also, knowing that the operation of searching a block in the blockchain is feasible in a sequential method (block by block). To verify if the position of the block that owns the parameters needed for the authentication may induce to a additional cost, we use a blockchain having 1000 blocks and for both scenarios, we measure the time needed for the authentication in the following cases: (1) the parameters needed for the authentication are in the first block of the blockchain; (2) the parameters needed for the authentication are in the block 500 of the blockchain; and (3) the parameters needed for the authentication are in the last block of the blockchain (1000).

B. Evaluation framework

In order to evaluate the time consumption of our approach, we used two Virtual Machines (VMs) as end nodes. The first VM was designed as the gateway that hosts the blockchain and the second as a smart thing. Table I describes their features.

The development of the authentication approach's steps was provided using *Java* language.

CPU architecture	CPU operation mode	CPU max speed	RAM	Operation System
x86_64	32-bits	2 GHz	256 MB	Debian 7.8

TABLE I: Experimentation VMs' features

We are aware about the difference of performance between VMs and some of the common smart objects. We are also aware about the fact that the obtained results depends on the used language (Java) and are different when using other languages (e.g., *C or C++*). Nonetheless, the goal of our evaluation is the evaluation of our approach's additional cost in comparison to classical methods. Consequently, the comparison is fair since all the protocol's operations are realized on the same basis, language and material.

[2]Any other hash algorithm can be used, and this according to the hosting system's requirements, needs and capacity

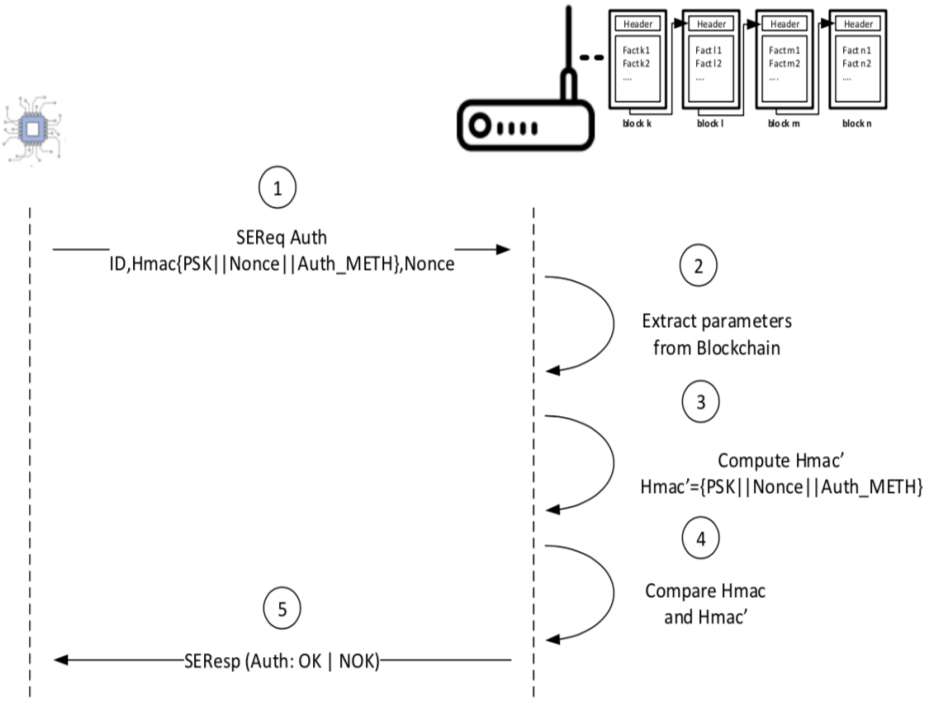


Fig. 2: Adaptive authentication approach: PSK use case

C. Evaluation results

1) Security and performance requirements evaluation: Our approach is an adaptive scheme that allows the usage of other authentication methods. Thus, some of the security features and robustness such as integrity, confidentiality, Non repudiation, robustness against replay and spoofing attack, protection against sybil attacks, etc. depends only on the chosen scheme. Ergo, in this section we focus on the evaluation of the security and performance features discussed earlier and that must be satisfied by an adaptive authentication approach:

Scalability: our scheme relies on a blockchain, which, in turn, relies on a peer-to-peer network. It is known that peer-to-peer networks are one of the best solutions to meet scalability at large scale [3][30].

Mobility of nodes: in our approach, the parameters related to the authentication and the authorization are stored in a blockchain. Since the blockchain is a decentralized system, all the gateways which are the peers that host the blockchain, host an updated version of the latter and which contains all the parameters needed to the authentication. Thus the nodes' mobility does not represent any obstacle, since any gateway can read the blockchain.

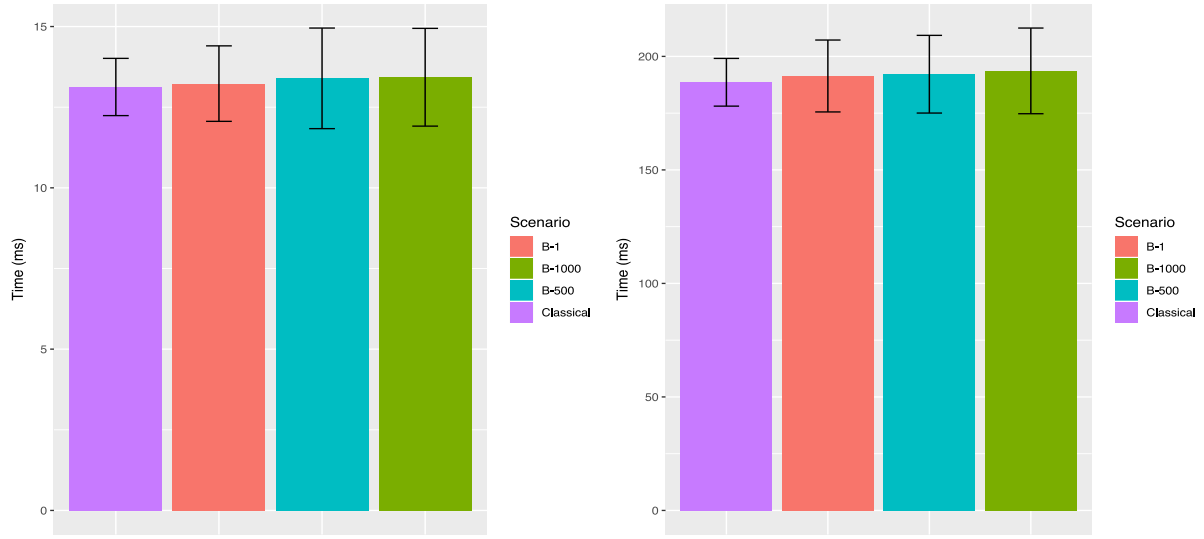
Heterogeneity of supported approaches: our scheme represents a way to extract and send the parameters needed and used by other schemes. Any method's parameters can be stored, thus, our approach supports the use of a multitude of other schemes.

Initialization phase: in order to add a new device, the user needs only to add the parameters needed for the authentication and authorization in the blockchain, which does not require any physical intervention on the gateways. Hence, we propose a very lightweight initialization approach, that brings numerous savings in comparison to other methods.

Robustness against cryptanalysis attacks: our scheme uses the other methods as they are designed. Thus, the robustness to cryptanalysis attacks are exactly the same as in classical cases. Nevertheless, we proposed a customized SAReq (described in Section III-C) where only a hash (HMAC) on the parameters is sent. Therefore, an attacker that intercepts this data, cannot reconstruct the original data, since hash algorithms are injective functions.

2) Numerical results: In this section we present the numerical results obtained upon the time consumption evaluation of our approach. As described in Section IV-A, for each use case (authentication with PSK and mutual authentication with certificate) we measured 4 values of the authentication time: (1) without our approach, which we present as *Classical case* in the further results; (2) the parameters needed for the authentication are in the first block of the blockchain, presented as *B-1*; (3) the parameters needed for the authentication are in the block 500 of the blockchain, presented as *B-500*; and (4) the parameters needed for the authentication are in the last block of the blockchain, presented as *B-1000*.

The Figure 3.a exhibits the results obtained from the experimentation of the first use case: authentication using PSK. Each result describes the average obtained through the execution of the same scenario **200 times**. Without surprise, the *Classical* method realizes the best authentication time with 13.13 milliseconds (ms) and a standard deviation of 0.88 ms. Then, closer is the needed block in the blockchain, better is



(a) (b)

Fig. 3: Authentication time needed: (a) PSK use case; (b) Mutual authentication with certificates

the time needed for the authentication, getting the following times: 13.23 ms, 13.39 ms and 13.43 ms, respectively for the scenarios *B-1*, *B-500* and *B-1000*. Also, the standard deviations obtained are very narrow with respectively 1.17 ms, 1.56 ms and 1.52 ms. One can note that the additional cost caused by our approach is negligible.

The Figure 3.b describes the results of the second use case: mutual authentication using certificates. As in the last experimentation, each result represents the average obtained through the execution of the same scenario 200 times. We obtained the same ranking of the last experimentation for the needed time to realize the mutual authentication with 188.60 ms, 191.37 ms, 192.15 ms and 193.62 ms, respectively for the *Classical*, *B-1*, *B-500* and *B-1000* scenarios. Similarly to the last use case, the standard deviations obtained are very narrow with respectively 10.51 ms, 15.84 ms, 17.11 ms and 18.87 ms. In this case also, the additional cost caused by our approach is insignificant.

Finally, the time needed to find the block in the blockchain depends on the blockchain used. For exemple, the time needed to look for a transaction in Bitcoin is not the same as for another blockchain less used, due to the number of blocks and transactions stored. However, optimization research approaches can be applied.

Regarding the financial cost related to the blockchain use, there are two cases: (1) if a private blockchain is used, then, there is no need for transaction payment. (2) if a public blockchain is used than, for each device, only one transaction is needed during the initialization phase and which contains the parameters that must be stored in the blockchain. Then, during the system's functioning, the gateways do only reading operations from the blockchain, which are free cost operations. The cost of one transaction depends on the blockchain used. However, it remains, generally, an insignificant cost. For example, if we consider *Ethereum Classic* blockchain. The cost of the transaction[3] is about 0.058 *EURO* as computed by Equation 1 as it was provided in [3]. Moreover, according to studies like [31] and [32], the evolution of the cryptocurrencies rates will get more stable over time. Even better, *Ethereum* developers and community are working on regulating and stabilizing the amounts of fees related to smart contracts use[4] [3].

$$\begin{aligned}\text{Cost} = \text{Number of Transactions} \times \text{Transaction cost}\\ \text{in } \textit{Gas} \times \textit{Gas} \text{ cost in } \textit{ETC} \times \textit{ETC} \text{ cost in } \textit{EURO}\end{aligned} \quad (1)$$

Consequently, considering these experimentations and results, one can conclude about the lightness and the low cost of our approach, while bringing more flexibility to the authentication system.

V. Conclusion and Future works

Despite the numerous works aiming the proposal of new secure approaches and schemes for IoT, still the majority of works does not satisfy all the system's needs and requirements such as allowing the mobility of nodes, the usage of numerous heterogeneous methods or the easy integration of new devices and services. Hence, in this work we propose an adaptive approach that ensures the authentication and the authorization for IoT devices. Our approach relies mainly on the use of a blockchain, which makes it completely decentralized. Furthermore, it satisfies all the discussed requirements. We

[3]The considered ETC value during the writing of this paper (August 2018) is 1 ETC = 11.67 *EURO*

[4]https://smartereum.com/6777/buterin-expresses-concern-over-stabilizing-ethereum/.

provided a real implementation of our approach and the results of its evaluation showed clearly its lightness and low cost.

This paper describes only the first phase of approach proposal which is related to the exchange protocol between the devices and the gateways. Thus, in our future work, we will describe the second part of the proposal which concerns the data storage in the blockchain, how it is protected and secured knowing that any user can read blockchain's block, and finally how the gateways can read this secured information.

REFERENCES

[1] Gartner Says By 2020, More Than Half of Major New Business Processes and Systems Will Incorporate Some Element of the Internet of Things. Technical report, Gartner, Inc, 2016.

[2] Badis Hammi, R Khatoun, Sherali Zeadally, Achraf Fayad, and Lyes Khoukhi. Internet of Things (IoT) Technologies for Smart Cities. *IET Networks*, 2017.

[3] Mohamed Tahar Hammi, Badis Hammi, Patrick Bellot, and Ahmed Serhrouchni. Bubbles of Trust: A decentralized blockchain-based authentication system for IoT. *Computers & Security*, 78:126–142, 2018.

[4] Meiyi Ma, S Masud Preum, W Tarneberg, Moshin Ahmed, Matthew Ruiters, and John Stankovic. Detection of runtime conflicts among services in smart cities. In *Smart Computing (SMARTCOMP), 2016 IEEE International Conference on*, pages 1–10. IEEE, 2016.

[5] Rajeev Alur, Emery Berger, Ann W Drobnis, Limor Fix, Kevin Fu, Gregory D Hager, Daniel Lopresti, Klara Nahrstedt, Elizabeth Mynatt, Shwetak Patel, et al. Systems computing challenges in the internet of things. *arXiv preprint arXiv:1604.02980*, 2016.

[6] Amir-Mohammad Rahmani, Nanda Kumar Thanigaivelan, Tuan Nguyen Gia, Jose Granados, Behailu Negash, Pasi Liljeberg, and Hannu Tenhunen. Smart e-health gateway: Bringing intelligence to Internet-of-Things based ubiquitous healthcare systems. In *Consumer Communications and Networking Conference (CCNC), 2015 12th Annual IEEE*, pages 826–834. IEEE, 2015.

[7] Emmanouil Vasilomanolakis, Jörg Daubert, Manisha Luthra, Vangelis Gazis, Alex Wiesmaier, and Panayotis Kikiras. On the security and privacy of Internet of Things architectures and systems. In *Secure Internet of Things (SIoT), 2015 International Workshop on*, pages 49–57. IEEE, 2015.

[8] Haodong Wang, Bo Sheng, Chiu C Tan, and Qun Li. Comparing symmetric-key and public-key based security schemes in sensor networks: A case study of user access control. In *The 28th international conference on distributed computing systems*, pages 11–18. IEEE, 2008.

[9] Konstantinos Christidis and Michael Devetsikiotis. Blockchains and smart contracts for the internet of things. *IEEE Access*, 4:2292–2303, 2016.

[10] Mohamed Tahar Hammi, Patrick Bellot, and Ahmed Serhrouchni. BC-Trust: A decentralized authentication blockchain-based mechanism. In *Wireless Communications and Networking Conference (WCNC), 2018 IEEE*, pages 1–6. IEEE, 2018.

[11] Hitesh Malviya. How Blockchain will Defend IOT. 2016.

[12] Kai Zhao and Lina Ge. A survey on the internet of things security. In *Computational Intelligence and Security (CIS), 2013 9th International Conference on*, pages 663–667. IEEE, 2013.

[13] Yuchen Yang, Longfei Wu, Guisheng Yin, Lijie Li, and Hongbin Zhao. A survey on security and privacy issues in Internet-of-Things. *IEEE Internet of Things Journal*, 4(5):1250–1258, 2017.

[14] Kim Thuat Nguyen, Maryline Laurent, and Nouha Oualha. Survey on secure communication protocols for the Internet of Things. *Ad Hoc Networks*, 32:17–31, 2015.

[15] Namje Park, Marie Kim, and Hyo-Chan Bang. Symmetric key-based authentication and the session key agreement scheme in IoT environment. In *Computer Science and its Applications*, pages 379–384. Springer, 2015.

[16] Mohamed Tahar Hammi, Erwan Livolant, Patrick Bellot, Ahmed Serhrouchni, and Pascale Minet. A lightweight IoT security protocol. In *Cyber Security in Networking Conference (CSNet), 2017 1st*, pages 1–8. IEEE, 2017.

[17] Riccardo Bonetto, Nicola Bui, Vishwas Lakkundi, Alexis Olivereau, Alexandru Serbanati, and Michele Rossi. Secure communication for smart iot objects: Protocol stacks, use cases and practical examples. In *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2012 IEEE International Symposium on a*, pages 1–7. IEEE, 2012.

[18] Thomas Kothmayr, Corinna Schmitt, Wen Hu, Michael Brünig, and Georg Carle. DTLS based security and two-way authentication for the Internet of Things. *Ad Hoc Networks*, 11(8):2710–2723, 2013.

[19] Francisco Vidal Meca, Jan Henrik Ziegeldorf, Pedro Moreno Sanchez, Oscar Garcia Morchon, Sandeep S Kumar, and Sye Loong Keoh. HIP security architecture for the IP-based internet of things. In *Advanced Information Networking and Applications Workshops (WAINA), 2013 27th International Conference on*, pages 1331–1336. IEEE, 2013.

[20] D Cooper, S Santesson, S Farrell, S Boeyen, R Housley, and W Polk. RFC 5280: Internet X. 509 Public Key Infrastructure Certificate and CRL profile. *Internet Engineering Task Force (IETF), May*, 2008.

[21] Tim Dierks and Eric Rescorla. RFC 5246: The Transport Layer Security (TLS) Protocol Version 1.2. *Internet Engineering Task Force (IETF) , August*, 2008.

[22] Rescorla E and Modadugu N. RFC 6347: Datagram Transport Layer Security Version 1.2. *Internet Engineering Task Force (IETF) , January*, 2012.

[23] Ali Dorri, Salil S Kanhere, Raja Jurdak, and Praveen Gauravaram. Blockchain for iot security and privacy: The case study of a smart home. In *Pervasive Computing and Communications Workshops (PerCom Workshops), 2017 IEEE International Conference on*, pages 618–623. IEEE, 2017.

[24] Ali Dorri, Salil S Kanhere, and Raja Jurdak. Blockchain in Internet of Things: challenges and solutions. *arXiv preprint arXiv:1608.05187*, 2016.

[25] Thomas Hardjono and Ned Smith. Cloud-based commissioning of constrained devices using permissioned blockchains. In *Proceedings of the 2nd ACM International Workshop on IoT Privacy, Trust, and Security*, pages 29–36. ACM, 2016.

[26] Mohamed Tahar Hammi, Erwan Livolant, Patrick Bellot, Ahmed Serhrouchni, and Pascale Minet. A lightweight mutual authentication protocol for the iot. In *International Conference on Mobile and Wireless Technology*, pages 3–12. Springer, 2017.

[27] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. 2008.

[28] Seyoung Huh, Sangrae Cho, and Soohyung Kim. Managing iot devices using blockchain platform. In *Advanced Communication Technology (ICACT), 2017 19th International Conference on*, pages 464–467. IEEE, 2017.

[29] Danny Dolev and Andrew Yao. On the security of public key protocols. *IEEE Transactions on information theory*, 29(2):198–208, 1983.

[30] Eng Keong Lua, Jon Crowcroft, Marcelo Pias, Ravi Sharma, and Steven Lim. A survey and comparison of peer-to-peer overlay network schemes. *IEEE Communications Surveys & Tutorials*, 7(2):72–93, 2005.

[31] Kenji Saito and Mitsuru Iwamura. How to Make a Digital Currency on a Blockchain Stable. *arXiv preprint arXiv:1801.06771*, pages 1–15, 2018.

[32] Ousmène Jacques Mandeng. Cryptocurrencies, Monetary Stability and regulation. Technical report, 2018.