Research Article

A Lightweight Blockchain Framework for IoT Integration in Smart Cities

Dlimi Zakariae[a*], Ezzati Abdellah[b], Ben Alla Saïd[c]
[a*]LAVETE Laboratory, Faculty of Science and Technology, Hassan First University of Settat, Morocco.
E-mail: zakariae.dlimi@gmail.com
[b]LAVETE Laboratory, Faculty of Science and Technology, Hassan First University of Settat, Morocco.
[c]LAVETE Laboratory, Faculty of Science and Technology, Hassan First University of Settat, Morocco.

Article History: Received: 11 January 2021; Accepted: 27 February 2021; Published online: 5 April 2021

Abstract: Smart cities heavily rely on technological enablers for their success, and more specifically on the IoT. This network, which will reach billions of components, now suffers from several constraints. In recent research, the blockchain has been proposed to provide answers on the limits of the centralized model, and on security. The integration of blockchain and IoT, however, still has issues that are being resolved, which are energy consumption, computing capacity, and storage capacity, due to the low capabilities of connected objects. In this article, we present our lightweight framework that we implemented for the integration of blockchain and IoT, and simulating on machines close to the configuration of the majority of current connected objects.

Keywords: IoT, Blockchain, Smart City.

1. Introduction

The IoT is a real accelerator for the development and success of smart cities, allowing it to migrate all components of the city to the digital world by connecting these physical objects and endowing them with digital capabilities. Connected objects will reach billions in 2020 according to Gartner [1], and they have low, to medium, then high capacities; and they are provided by different suppliers. These factors pose the problems of network bottleneck, adapting to advanced standards of security, standardization, and homogeneity [2]. Blockchain technology provides solutions for several fields of application through its decentralization, disintermediation, transparency, and security features [3]. Its application integrated with IoT has been the subject of varied research, including Smart Transportation [4, Smart Home [5], Smart healthcare [6], and smart agriculture [7]. The constraints encountered during this integration are mainly: energy consumption, computing capacity, and storage capacity to access and participate in the blockchain network with devices, the majority of which are of limited capacity.

The different proposed implementations can be classified under the following three categories of architecture models [8]: (i) The IoT to IoT architecture, whose devices can communicate with each other and then communicate with the blockchain network via capable devices (Figure 1).

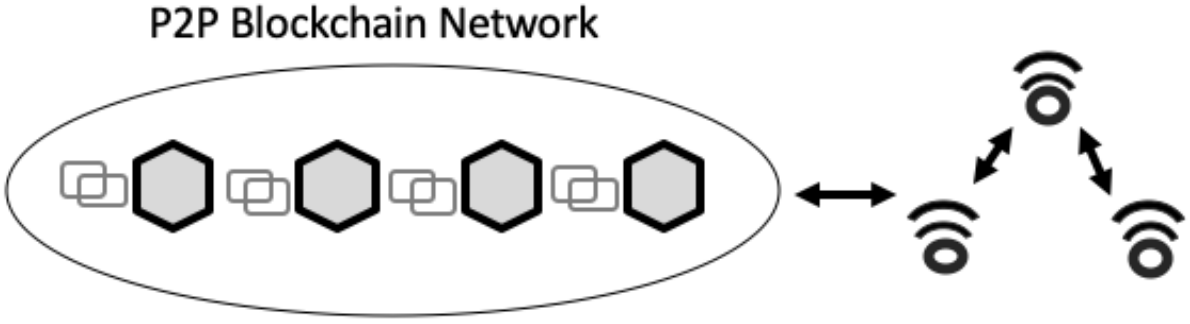

Figure 1. IoT to IoT Architecture

This pattern ensures fast communication without requiring high technical capacity, but the network does not fully comply with security and compliance requirements. (ii) The IoT to Blockchain architecture, where all devices are connected to the blockchain network to ensure privacy, compliance and security (Figure 2).

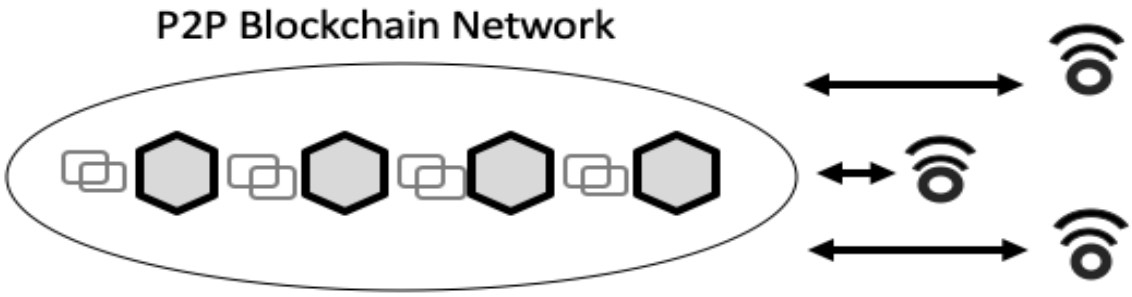

Figure 2. IoT to Blockchain architecture

On the other hand, all devices must be able to participate in the blockchain in terms of resources. (iii) And finally, the IoT to Blockchain model via Cloud / Fog network, which takes advantage of the capabilities of Fog and Cloud computing to perform the encryption and compression functionalities (Figure 3).

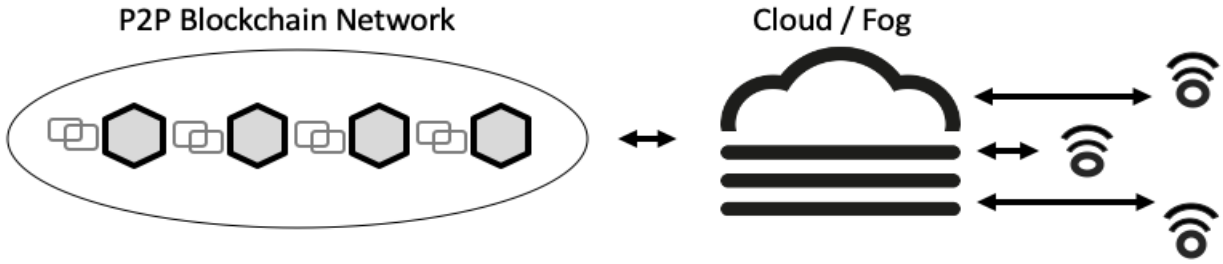

Figure 3. IoT to Blockchain via Cloud / Fog architecture

In this paper, we propose a lightweight blockchain framework, which brings decentralization, security and disintermediation to the IoT. Initially, we put in place the main operating bricks of a Blockchain, with the necessary interfaces for IoT devices, all implemented by the Node.js framework.

2. Proposed Framework

Our aim is to arrive at a lightweight blockchain framework for computing and processing data from a consumer point of view, integrated with IoT to meet the expectations of smart cities. This starting model that we are proposing is part of both an IoT to IoT and IoT to Blockchain architecture, in which IoT devices participate in the blockchain network, and other devices are connected directly to these blockchain devices when their capacity is not sufficient. Node.js was chosen as the framework for implementing the blockchain application of the participating nodes. This framework is known for its richness in development tools, and for its few requirements in terms of machine resources. The consensus implemented in this initial model is the Proof of Work, a security and integrity mechanism implemented by Bitcoin. Since this mechanism is known for its very high computing capacity requirements, we have chosen the validation of the block by the Binary Hash algorithm instead of the Hash, which is less resource intensive. This consensus has been put in place in a flexible manner, in order to easily replace it with another more efficient mechanism, depending on the context of application.

2.1. Architecture

The model we offer allows devices in the IoT network to connect to the Blockchain network. We define 3 roles for a device to participate in the IoT and Blockchain environment. (i) The Node Validator role, where the device holds its key pair, stores the Blockchain register locally, creates signed transactions, and participates in the validation of newly created blocks according to the consensus. (ii) The role of Node Wallet Only, whose device does not participate in the validation of blocks, but can create signed transactions, as well as storing the blockchain ledger locally. This role is justified for devices that do not have sufficient computing capacity, or sufficient battery power, or timely availability to demonstrate proof of consensus work. (iii) The third role is Thing Only, whose device does not participate in block validation, does not have a key pair and therefore cannot create signed transactions. On the other hand, it can store the ledger of the blockchain locally by synchronizing itself with the other nodes; through it cannot store it, it solicits the reading and writing from the other nodes with the roles of Node Validator or Node Wallet Only, via the interfaces implemented in our architecture. We can say that it is a hybrid architecture between the IoT to the IoT and the IoT to the Blockchain (Figure 4).

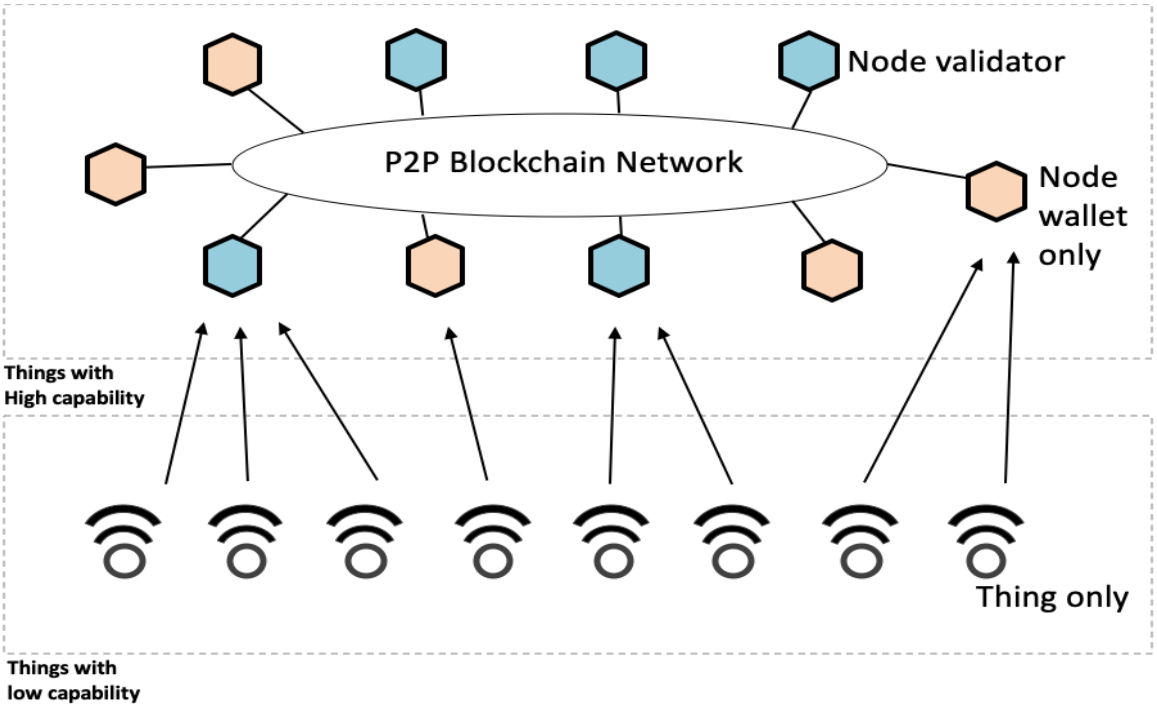

Figure 4. Proposed framework's architecture

2.2. Blockchain Node's Components

The blockchain node is an application that can play the role of Node Validator or Node Wallet Only. It consists of a Blockchain Application in order to provide the main functionalities, namely the storage of the ledger, the security keys, the signature, the control and the validation, and the consensus mechanism (Figure 5). The nodes form the blockchain network by connecting to each other through the Peer to Peer Websocket interface. The blockchain node also implements the connection interfaces with connected objects, the most used at the moment and which are: MQTT, CoAP, and REST API.

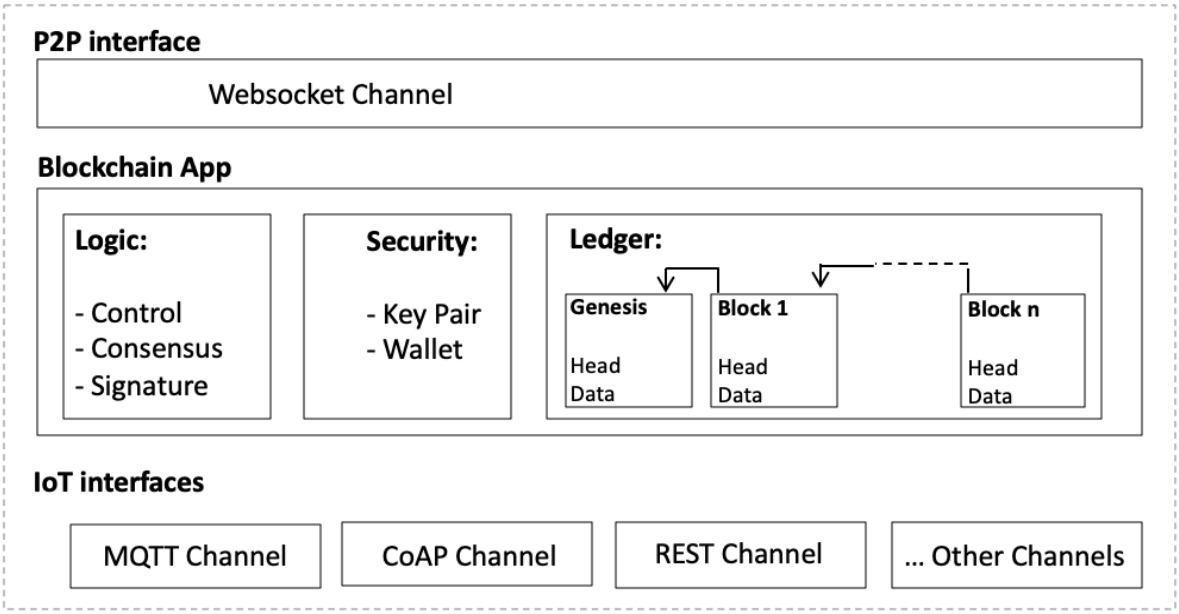

Figure 5. Proposed Framework's Components

2.3. Logic

The process of running the solution from starting a node is described by Algorithm 1, shown below:

Algorithm 1 How the solution is working at startup

1. Starting the node
2. Creation of the Genesis block
3. Initiation or receipt of transaction creation order
4. Creation and signature of a new transaction
5. Adding the transaction to the Transaction Pool
6. Broadcast transaction
7. On receipt of transaction, compliance control. If Crypto-currency system, check balance
8. If condition (time or size completed), creation of block by Node Validator
A. Adding Reward Transaction
B. Block validation (For PoW: Difficulty adjustment and Binary Hash calculation)
C. Adding the block to the local ledger
D. Broadcast the ledger
9. Upon receipt of a ledger, check and replace the local ledger if it is smaller

The pseudo code of the block validation logic according to the used PoW consensus is described below:

Pseudo Code 1 PoW Block validation

```
Input: lasgBlock, Data
Output: Block{timestamp, lastHash, hash, data, nonce, difficulty}
Declare: hash, timestamp, nonce, difficulty
Initialize: nonce := 0, difficulty := lastBlock->difficulty
1   Do
2   nonce := nonce+1
3   timestamp := currentTime()
4   difficulty := adjustDifficulty(lastBlock, timestamp)
5   hash := hash(timestamp, lastHash, data, nonce, difficulty)
6   While (Substring(hash, 0 to difficulty) != Repeat ('0', difficulty))
```

As described by the PoW consensus, the difficulty of performing proof of work increases with the growth in the overall processing capacity of the Blockchain network. The adjustment procedure has been implemented by using the pseudo code described below:

Pseudo Code 2 Difficulty adjustment

```
Input: lastBlock, timestamp
Output: difficulty
Declare: difficulty, HASH_RATE
Initialize: difficulty := lastBlock->difficulty, HASH_RATE = 5 minutes
1    If lastBlock->timestamp + HASH_RATE > currentTime
2    difficulty := difficulty +1
3    Else
4    difficulty := difficulty - 1
```

The pseudo code of the logic of the local ledger replacement of a node according to the PoW consensus is described below:

Pseudo Code 3 Ledger replacement

```
Input: newChain
Output: Boolean
1   If length(newChain) <= length(existing chain)
2   Return false
4   If newChain[0] != block genesis
5   Return false
6   Loop for all blocks in the new chain
7   If (block->lastHash != lastBlock->hash) OR (block.hash != hash(block-> {timestamp, lastHash, data, nonce,
    difficulty})
8   Return false
9   Return true
```

2.4. Simulation and Results

We simulated our proposed framework on a blockchain made up of 10 nodes and a connected objects simulator. Each node of the blockchain is a server under Amazon Linux 2 AMI, configured with a minimum of necessary resources close to an IoT component such as Raspberry Pi 2. The nodes of the blockchain are *t2.micro* instances (Variable ECU, 1 vCPU, 2.5 GHz, Intel Xeon Family, 1 GB memory, EBS only). We simulated the interactions of connected objects by Curl requests in Shell Bash scripts, which send data to the nodes of the blockchain through the REST API interface. We ran a transaction recording scenario on the different nodes, with a block validation triggered every 5 minutes. Since the computing capacity of the entire blockchain remained the same during the simulation phase, the difficulty of PoW stagnated in 4.

We recorded the CPU and RAM consumption of the Node process of the blockchain application by using the System information 4.0 module, during the validation phase of the block, and in the replacement phase of the local chain during the synchronization between nodes. This moment of replacing the local chain by the node is important from our point of view, because we will base our future work on its improvement, with a new model of organization and prioritization of data.

In the graph (Figure 6), we show the CPU usage per Node Blockchain process, during block validation. There is a frequent exceeding of the capacity of the resources initially allocated for the node, going up to 500%, by drawing available reserves in the cloud. The calculation of Binary hash responding to the difficulty of PoW has also been shown to be CPU resource intensive, and in the case of a hardware component with inelastic and limited resources, like the AWS Cloud, the model will arrive at its limits without being able to offer good performance. RAM usage remained between 7% and 10%.

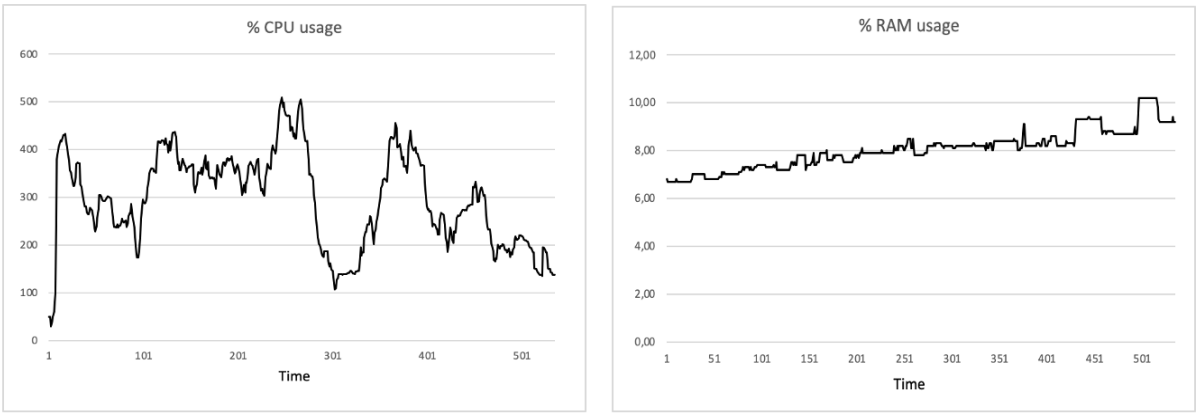

Figure 6. PoW Consensus CPU & RAM usage report.

For the chain synchronization function after block validation, the graphs (Figure 7) show stable and low consumption of CPU and RAM. This is because of the low degree of complexity of the synchronization function.

This will not be the case when the size of the blockchain register is very large. For the validation of the contents of the register, each node will have to recalculate a large number of hashes in the order of the size of the chain. The nodes must also all have a large storage capacity when the amount of data is important.

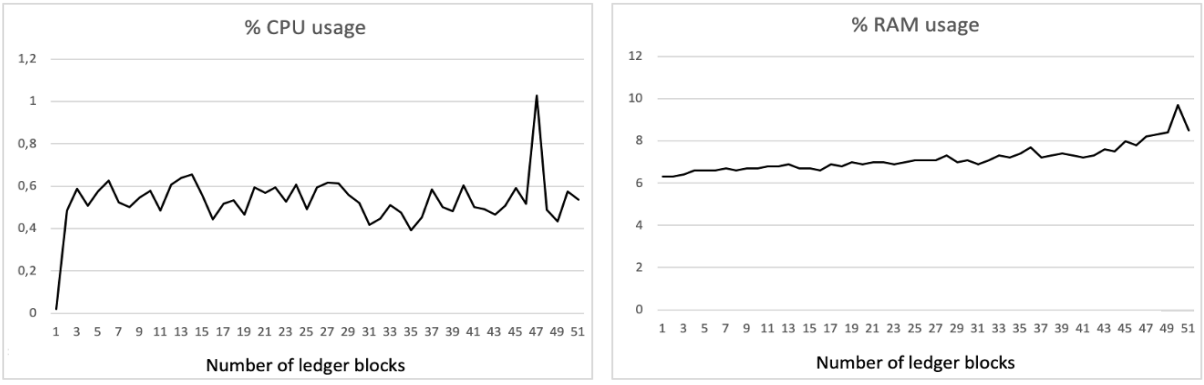

Figure 7. PoW Consensus CPU & RAM usage report with chain synchronization

3. Conclusion

In our current work, we proposed an initial lightweight blockchain framework to meet the constraints of IoT objects, and we implemented it to verify these limits with the PoW consensus. The only implementation of a blockchain with the node.js framework is not sufficient with the PoW binary hash based consensus, and requires optimization at the level of the consensus mechanism and the ledger storage mechanism.

Our future work aims to improve this initial framework with a consensus model and data storage more suited for the context of IoT with limited resources.

References

1. Rivera, J., & Van Der Meulen, R. (2016). *Forecast alert: internet of things—endpoints and associated services, worldwide.*
2. Čolaković, A., & Hadžialić, M. (2018). Internet of Things (IoT): A review of enabling technologies, challenges, and open research issues. *Computer Networks, 144,* 17-39. http://dx.doi.org/10.1016/j.comnet.2018.07.017
3. Song, J.C., Demir, M.A., Prevost, J.J., & Rad, P. (2018). Blockchain design for trusted decentralized IoT networks. In 2018 13th *Annual Conference on System of Systems Engineering (SoSE),* 169-174. http://dx.doi.org/10.1109/sysose.2018.8428720
4. Sharma, P.K., Moon, S.Y., & Park, J.H. (2017). Block-VN: A distributed blockchain based vehicular network architecture in smart City. *Journal of information processing systems, 13*(1), 184–195. https://doi.org/10.3745/JIPS.03.0065
5. Dorri, A., Kanhere, S.S., Jurdak, R., & Gauravaram, P. (2017). Blockchain for IoT security and privacy: The case study of a smart home. In 2017 *IEEE international conference on pervasive computing and communications workshops (PerCom workshops),* 618-623. http://dx.doi.org/10.1109/percomw.2017.7917634
6. Sahoo, M., Singhar, S.S., & Sahoo, S.S. (2020). A blockchain based model to eliminate drug counterfeiting. In *Machine Learning and Information Processing,* 213-222). Springer, Singapore. http://dx.doi.org/10.1007/978-981-15-1884-3_20
7. Lin, J., Shen, Z., Zhang, A., & Chai, Y. (2018). Blockchain and IoT based food traceability for smart agriculture. In *Proceedings of the 3rd International Conference on Crowd Science and Engineering,* 1-6. http://dx.doi.org/10.1145/3265689.3265692
8. Reyna, A., Martín, C., Chen, J., Soler, E., & Díaz, M. (2018). On blockchain and its integration with IoT. *Challenges and opportunities. Future generation computer systems, 88,* 173-190. http://dx.doi.org/10.1016/j.future.2018.05.046