Secure distributed ledgers to support IoT technologies data

Concepción-León, A.[1] and Endler, M.[1]

[1]Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio).
Rio de Janeiro, Brasil

{aleon@inf.puc-rio.br} endler@inf.puc-rio.br

Abstract. *Blockchain and Tangle are data structures and protocols used to create an immutable public record of data insured by a network of peer-to-peer participants that maintains a monotonic growing set of distributed data records known as ledgers. Both technologies provide a decentralized solution that can guarantee the exchange, of large amounts of valid and complete messages in the Internet of Things. This encrypted and secure peer-to-peer messaging mechanism is adopted in this project to manage the processing of IoT transactions. To maintain transactions private, and secure, the distributed consensus algorithms are responsible for validating and choosing transactions and recording them in the global ledger. Experimental results showed that the latency imposed by the consensus algorithms can have a negative effect on the online creation of reliable stories that track the events of the IoT networks. By using Complex Event Processing as a pre-processing step that allows selecting only those high level events, it is possible to obtain a significant improvement of latency in many situations. The main contribution is a middleware service that provides a framework for the construction of large-scale IoT applications that combine Complex Events Processing and different decentralized ledgers such as the blockchain of Ethereum or IOTA Tangle, for secure data storage.*

1. Introduction

The "Internet of Things" (IoT) enables the integration of the physical world with the virtual world of the Internet [Haller 2010]; and already has a strong impact in several areas such as smart homes and cities, environmental monitoring, asset monitoring, logistics, etc. Thanks to rapid advances in underlying technologies, IoT is opening tremendous opportunities for a large number of novel applications that promise to improve the efficiency of several economic sectors as well as the quality of our lives [Xia 2012].

Deployments of IoT systems in several fields entail an enormous increase in the collection and analysis of a large amount of data, by connecting multiple smart devices with the cloud-hosted services; therefore, IoT will require to handle an increasingly large volume of data/information in real time from a variety of sources and to detect significant events of the environment and/or from mobile entities by correlating this data in real time. Many of these collected data are valuable only if one can ensure their integrity, confidentiality, privacy and non violability since this is necessary for ensuring the integrity of the derived information and the correct and accurate decision making.

Currently, many IoT devices, data transmission protocols and databases are supported by security models susceptible to accidentals and malicious attacks, including data tampering [Razzaq 2017]. Distributed ledgers offer an elegant solution to solve this

problem, since its decentralized, autonomous and trustful capabilities make it an ideal component of IoT solutions [Vitalik 2015]. The adoption of distributed ledgers with this important characteristic allows a secure peer-to-peer messaging mechanism to manage the processing of huge amounts of IoT transactions and the coordination between the devices [Vitalik 2015]. Companies with IoT solutions have quickly become one of the first to adopt distributed ledger technologies, but in addition to having a decentralized IoT architecture, it is also necessary to ensure that it can scale while maintaining all transactions secure and trustful.

Among the several available technologies for implementing distributed ledgers data storage systems, we use two different technologies: the Ethereum blockchain and the IOTA Tangle. Ethereum blockchain provides a persistent structure where data is stored in a reserved space in each transaction. Tangle is a blockless distributed ledger of the cryptocurrency platform called IOTA and was created as a cryptocurrency for the IoT industry [Popov 2016]. Despite these important characteristics, there are associated problems that must be resolved before a ledger based-storage system can be applied to IoT systems, such as: privacy of the sensor data and users-sensitive data, centralization of miners, vulnerability, immutability of smart contracts, limits on data storage, consensus and slow writing, among others.

In this project, a filtering technique will be addressed to solve the problems related to the scalability and performance of the distributed ledgers, to be applied to an IoT system. In order to mitigate the problems previously mentioned, the network can be structured with an architecture where some computational intelligence is located the closest possible to where data is collected. Thus, instead of sending directly all the sensor/status data collected by all endpoint-devices (smart things), they will be analyzed and pre-processed locally at “the edge/fog of the network” before being sent to the ledger. To analyze, filter and collect data from the devices of an IoT system, the middleware ContextNet [Endler 2018] was used as a transport layer. In addition, it allows using techniques of Fog Computing such as Complex Event Processing (CEP) and data buffer, for data filtering. By implementing this strategy as part of this project, we intended to reduce the throughput of data sent to the ledger locally to solve the growing problems of running out of network bandwidth as the number of nodes grows (i.e. scalability).

The aim of this project is the implementation of a secure transaction storage system using distributed ledgers as a middleware service for the ContextNet. For this reason, the implementation of a trustless peer-to-peer messaging protocol based on Distributed Ledgers for ContextNet is proposed to create a more secured transport protocol for IoT. This messaging protocol will be capable to scale while stored in a distributed way and maintain private, secure and trustful transactions.

2. Enabling Technologies

2.1 Distributed Ledgers

A Distributed ledger is an immutable public record to store large volume of transactions, which can significantly facilitate the creation of reliable network histories for tracking the actions of nodes in an IoT system. Formally, a distributed ledger system is immutable because it fulfills that: at each times t> 0 there is a data set Ledger(t) that describes the entire system history up to that point. In principle, a transaction is stored in all the nodes of

the network and all copies are perfect replicas. This ledger is incremental, in the sense that Ledger (t) ⊂ Ledger (s) for all t <s; that is, the data is added but never deleted. Distributed ledgers are also secured by a network of peer-to-peer (P2P) participants that maintain a continuously growing set of data records. Each transaction is verified and recorded by the consensus of the majority of the participants in the system. Once the transaction entered the block, that information can never be removed or modified again [Christidis and Devetsikiotis 2016].

Ethereum (ETH) [Vitalik 2015] is a decentralized platform with a built-in programming language that executes smart contracts on a customized blockchain, and allows to deploy Decentralized Applications (DApps) on top of it. Ethereum provides a decentralized peer-to-peer network where participants can exchange trusted messages. In the Ethereum blockchain the block is stored in a multi-level data structure named Merkle Patricia tree (trie) that provides a persistent data structure for mapping between arbitrary-length binary data (byte arrays) [Wood 2014].

IOTA Tangle [Popov 2016] is a distributed ledger for storing transactions with a directed acyclic graph (DAG) instead of a public string of blocks. In Tangle, all data is stored in distributed and trustlful nodes among the network, and this design guarantees the integrity of the data that is sent. Tangle is scalable, lightweight and makes possible to transfer value without any fees; so that devices can trade exact quantities of resources on-demand, and store data from sensors and dataloggers securely and verified on the ledger [Popov 2016].

2.2 ContextNet Middleware

ContextNet [Endler 2018] is a scalable “Internet of Mobile Things” (IoMT) middleware which provides context services for wide- and large-scale pervasive collaborative applications such as on-line monitoring or coordination of mobile entities' activities. In the ContextNet project, communication and context distribution capabilities are implemented in the Scalabe Data Distribution Layer (SDDL), while other services and extensions are built as software modules on top of this distribution layer. Together, these software modules form the ContextNet middleware [Talavera 2016].

2.3 CEP

Complex Event Processing (CEP) is a data stream processing technology which is able to recognize relevant patterns of events from multiple sources that later will be processed, providing capabilities of filtering, aggregation, correlation and analysis of events. The goal of CEP is to program rules which check for event properties, and event pattern of interest and implement responsive actions that are triggered whenever the antecedent is satisfied [Robins 2010].

2.4 Mobile Hub

The Mobile Hub (M-Hub) [Talavera 2015] is a general-purpose gateway that manages and connects smart and also mobile objects with different WPAN technologies such as BLE and Classic Bluetooth. The main function of the M-Hub serves as an intermediary between the mobile objects and the services in the SDDL core of the ContextNet architecture. This middleware is responsible for discovering and connecting the smart objects to the SDDL Core [Talavera 2016].

3. Implementation

3.1 Architecture

The multilayer architecture of the project can be physically separated by 3 tiers: middleware tiers, application tiers and ledger tiers (Figure 1). This three-tiers architecture allows the project to be easy to extend and scale, with distributed and decentralized programming as a basic principle. Each of the tiers has associated logical layers where the corresponding technologies and patterns are used.

To access the distributed ledgers through ContextNet, the DLedger service was implemented, which is responsible for connecting the IoT applications with the different immutable public records, and therefore, facilitates the storage of data in the ledgers. The DLedger service implemented is part of the ContextNet middleware and it is also presented at each of the gateways within the SDDL core network. This service will improve the middleware with new functions related to security and storage, allowing adaptation to environments where distributed applications require blockchain technologies (Figure 1). The service is available as an API and allows you to control all the functions of each ledger. For the implementation of the API, the Tangle and Ethereum technologies were consumed through their libraries in the Java versions: **Web3j** for Ethereum and **IOTA Java** for Iota.

The Application Tier consists of one or more instances of applications that consume specific services of ContextNet through the M-Hubs. In addition, the Ledger Tier is in charge of storing all the data that the sensors of the system emit. The way to store them is through transactions made in the network. Each transaction is an object that has a field where any type of data can be stored. This is the most basic layer of the proposed system and contains two different ledgers technologies and it is possible to choose which of the two technologies they prefer to store the data (Figure 1).

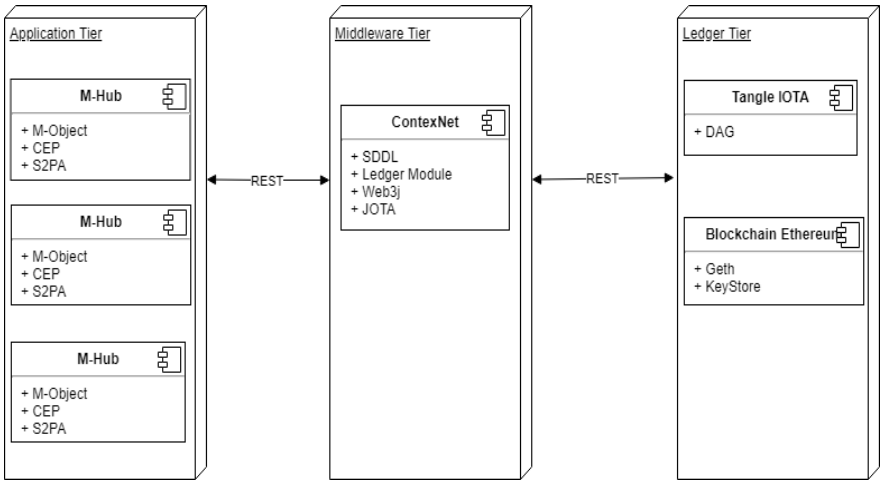

Figure 1. Diagram representing the three-tiers architecture of the system

3.2 Storage of context data using distributed ledgers

The context data was stored through transactions that include the recipient's address, the transaction data, payload and the transaction value. Transactions were broadcast to all participant nodes in the network and they were independently verified and "processed". Once transactions are buried under enough confirmations they can be considered irreversible [Bahga and Madisetti 2016]. Each transaction in IOTA consisted of 2673 trytes and the context data was stored as an encoded object in the field named “SignatureMessageFragment”.

For Ethereum transactions the context data were stored in unlimited arrays of bytes. Although any amount of data can be stored in these fields, there is a limit of data that can be stored in a block. In addition, the Ethereum network has a condition in which the sum of the transactions gas of the block (Tg), must not be greater than the GasLimit of the block. For this reason, it was necessary to calculate the gas limit of the transaction and ensure it does not exceed the gas limit of the block.

3.3 Data Reduction

The context data related to the system individually does not require much space, but if we reach a large volume of transactions with scales similar to the dimensions of a real IoT system, this becomes a problem. In addition, the consensus and slow writing of distributed ledgers affects the scalability and performance of the systems, which is an inconvenient to be adopted by the IoT systems. In order to reduce the amount of data inserted into the distributed ledgers and to face the scalability problem, some Fog computing techniques were developed.

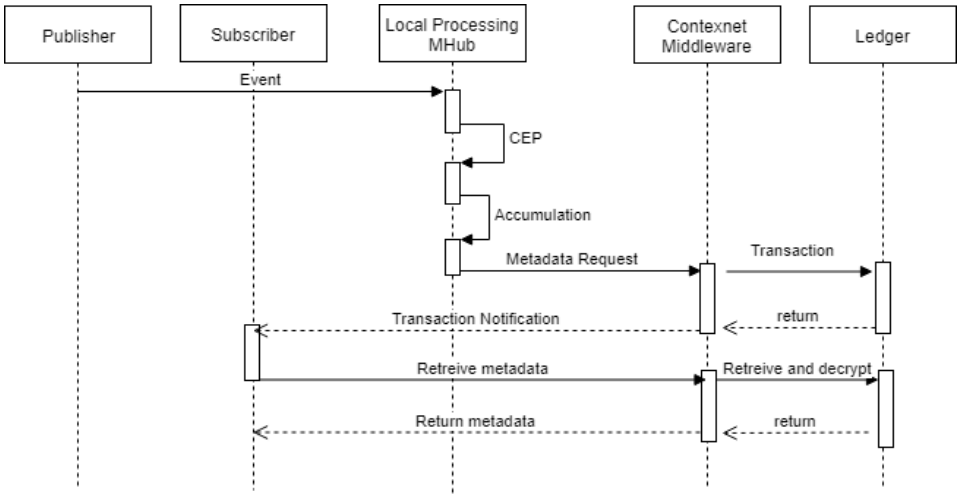

Figure 2. Sequence diagram representing the process of insertion in the leger with CEP and data buffering.

4. Performance Experiments and Results

To evaluate the system, an application that simulates the detection of people in different places within a building has been created. To achieve that, a total of 157 sensors that sent data for the IOTA and Ethereum network were simulated, and they can identify 150 possible people. Each people ("user") were detected inside the room using stations with

MHubs installed. The station proceeds to locate the person who is present in a short range radio, using Bluetooth action to obtain better accuracy. The data related to the user and the sensor metadata were stored in the general ledger distributed through a WiFi infrastructure.

The context data of the sensors (MObjects) were sent as a triple, and contains data related to the moment in which the event was executed (first element) in milliseconds, a specific value related to the identifier of the person who was detected, and the address within the network. The triple is stored in the IOTA and Ethereum transaction as previously described. The tests of both networks were carried out in environments with the same conditions.

To evaluate the system, three **(**3**)** possible environments or system configurations were analyzed. In the first configuration (Configuration 1), simple transactions were sent without using the accumulation technique or the CEP service. In this way, each sensor sent directly through ContextNet a transaction with the metadata corresponding to the event that was made. In the second configuration (Configuration 2), the data accumulation technique was taken into account. A set of metadata of the events are concatenated and sent forming a single transaction. After being sent, the process is repeated. This technique allows the accumulation of approximately 12 transactions without exceeding the block limit, being very important not to exceed this block limit. Finally, the third configuration (Configuration 3) also used the accumulation technique and CEP in the FOG computing, where a rule to filter the data has been established. Only people who have identifiers above a predetermined value, were detected. Thus, the system has special interest in this group of people, and any extra data that is obtained will simply be discarded.

An analysis of each configuration in both network (IOTA and Ethereum) was performed to evaluate different parameters, such as the number of events or actions that each sensor issued, which could end in transaction or not, and the number of transaction per sensor. The average of the time it takes to process each transaction per sensor (in milliseconds) and the values of the lowest (minimum) and highest (maximum) execution time of a transaction (of the total number of transactions) per sensor, were also analyzed. In addition, the number of pending transactions per instance of time was taking into account. The final parameter is the Fee Cost, which is the total fee that must be paid to the network so that the miners insert the transaction in the corresponding ledger.

For each configuration, the experiments were repeated three (3) times to make a comparison of the results. The instances of time analyzed in each configuration are defined as the exact moment (in milliseconds) in which a transaction was issued. A transaction is considered pending when it was started but has not yet been completed and inserted into the ledger. In the Ethereum network, when measurements of gas prices cross the gasLimit threshold, they are no longer selected. When this happens, the previous transactions are accumulated and a single transaction with the data previously obtained; and a new data accumulation period immediately begins. To test which environment performs best, each single node was tested on computers with the same conditions. The test environment is a computer with Windows 8.1, Core(TM) i7 1.80 GHz and 8 GB of RAM. The experiments were carried out in a period of time of 25 minutes.

4.1 Ethereum Network Evaluations

To evaluate the performance of the Ethereum Network, three (3) different configurations and six (6) different parameters were analyzed. The average number of events sent per sensor varied between 6.10 ± 0.64 for the first configuration and 26.50 ± 7.83 for the third (**Figure 3a**). The number of transactions for each configuration is less than the number of corresponding events, except in the first configuration, where both values have the same value (**Figure 3b**).

The evaluation of the number of pending transactions was made by analyzing, at each instant of time that a new transaction was executed, how many transactions had not yet been completed. As a result of the decrease in the number of transactions and the average execution time, it was also observed a smaller number of pending transactions in the configuration 3 and 2 in relation to the previous configurations, the first configuration being the one that showed the highest value (**Figure 3c**).

The analysis of the average of the time it takes for each transaction to be complete was also performed (**Figure 3c**), and its results can be correlated with the time value (in millisecond) of the transaction that took longer to execute (maximum) (**Figure 3e**), and the one that took less time (minimum) (**Figure 3f**). In general, it was possible to observe a decrease in the average running time of the Configuration 2 and 3 in relation to the first condition, with the Configuration 3 (simple transaction with data buffering and CEP) being the lowest value.

The cost per sensor was also analyzed (**Figure 3g**). This cost is measured in gas units and represents the fee that is paid to the miners for inserting the transactions into the blocks. Normal-sized transactions have a gas price of 90000. The total cost per sensor is directly related to the number of transactions, therefore, lower costs were observed in the configuration 3 and 2 in relation to the first, with the third one showing the lowest values.

4.2 IOTA Network Evaluations

To evaluate the performance of the IOTA network, the same configurations used in the Ethereum network were analyzed; except for the Fee Cost. Unlike the Ethereum network, in the IOTA network there is no need to pay a fee, so the total cost to pay to perform the transactions is 0. **Figure 3a** shows the results of the evaluation of the number of events per sensor in the three configurations previously mentioned. The average of transactions made per sensor in the first configuration is the same as the number of total events per sensor, while in the second and third configuration these values were slightly lower (**Figure 3b**).

When analyzing the average time it took for the set of transactions to be executed per sensor, the use of data accumulation by buffering data with simple transactions showed visible differences in the results related to the use of simple transactions in the IOTA network. . In this configuration 2, all 157 sensors emit the events and a buffer accumulates all the transactions and sends it in groups of 10. In the third configuration, a reduction of the operating time per sensor was also observed in relation to the previous configurations (**Figure 3c**). Finally, the number of pending transactions in each instance of time was also evaluated. In the first configuration, in each instant of time in which a new transaction was issued, there were approximately 417.55 ± 141.01 pending transactions. A lower number

of pending transactions was generally observed in the second configuration. The third configuration evaluates the behavior of the system using CEP from FOG computing, where a rule for filtering the data was previously established. The data accumulation using data buffering was also implemented. The use of these tools allowed decreasing the number of pending transactions in relation to the previous configurations, as observed in **Figure 3d**.

4.3 Ethereum and IOTA Network Comparison

The performance of the Ethereum and IOTA network in the three Configurations was compared, taking into account the 6 parameters previously mentioned. **Figure 3** shows the graphical representation of the performance of both networks, including the averages and standard deviations of the three independent experiments. All statistical analyses were carried out using GraphPad Prism version 6.00 for Windows (GraphPad Sofware, San Diego California USA). One-way ANOVA test was used to compare the differences among groups, with a confidence interval of 95% for all experiments.

In all the parameters evaluated, statistically significant differences between Configuration 3 values of both ledgers were observed. Lower values of the average time of execution and pending transactions were observed in the Configuration 3 of Ethereum, achieving a reduction of 89.66% and 37.48%, respectively, in relation to the same configuration in IOTA (**Figure 3c and 3d**). Also, similar values of reduction were detected between the Configuration 3 of Ethereum and IOTA, when comparing the average values Minimum (86.61%) and Maximum (91.4%) of the execution time of the transactions (**Figure 3e and 3f**). A reduction of 75.24% in the fee cost of Ethereum network was observed, comparing the Configuration 3 with the Configuration 1 values. In addition, in IOTA network there is no need to pay a fee, so the total cost to pay is 0 for all the configurations (**Figure 3g**).

To analyze the efficiency of accumulation of transactions, the percentage of reduction of sent transactions, in relation to the number of events, was analyzed (**Figure 3h**). The following formula was used:

$$\%\,reduction = \frac{No.events - No.transaction}{No.events} \times 100$$

where *No. events* represents the number of events (**Figure 3a**), and *No. transaction* the corresponding number of transaction (**Figure 3b**). In the Configurations 1 of both ledgers, simple transactions were performed without data buffer or CEP, so there was no accumulation, being a percentage of 0. The use of both, data buffer and CEP, increased the percentage of reduction in the number of transactions in both ledgers (Ethereum: 93.29% and IOTA: 97.62%), in relation to the Configuration 2 (Ethereum: 91.71% and IOTA: 92.48%), where data buffer was only used. The configuration 3 of IOTA is the one that showed the highest percentage of reduction of all the analyzed groups (**Figure 3h**).

In order to compare the efficiency of CEP and data buffer in Ethereum in relation to IOTA, the percentage of reductions of sent transactions was also analyzed. For that, the number of transactions sent per sensor, in relation to the number of total events received, was compared. Configuration 1 was used as a control, where no event was filtered and no transaction was accumulated. The use of data buffer and CEP in Tangle achieved the

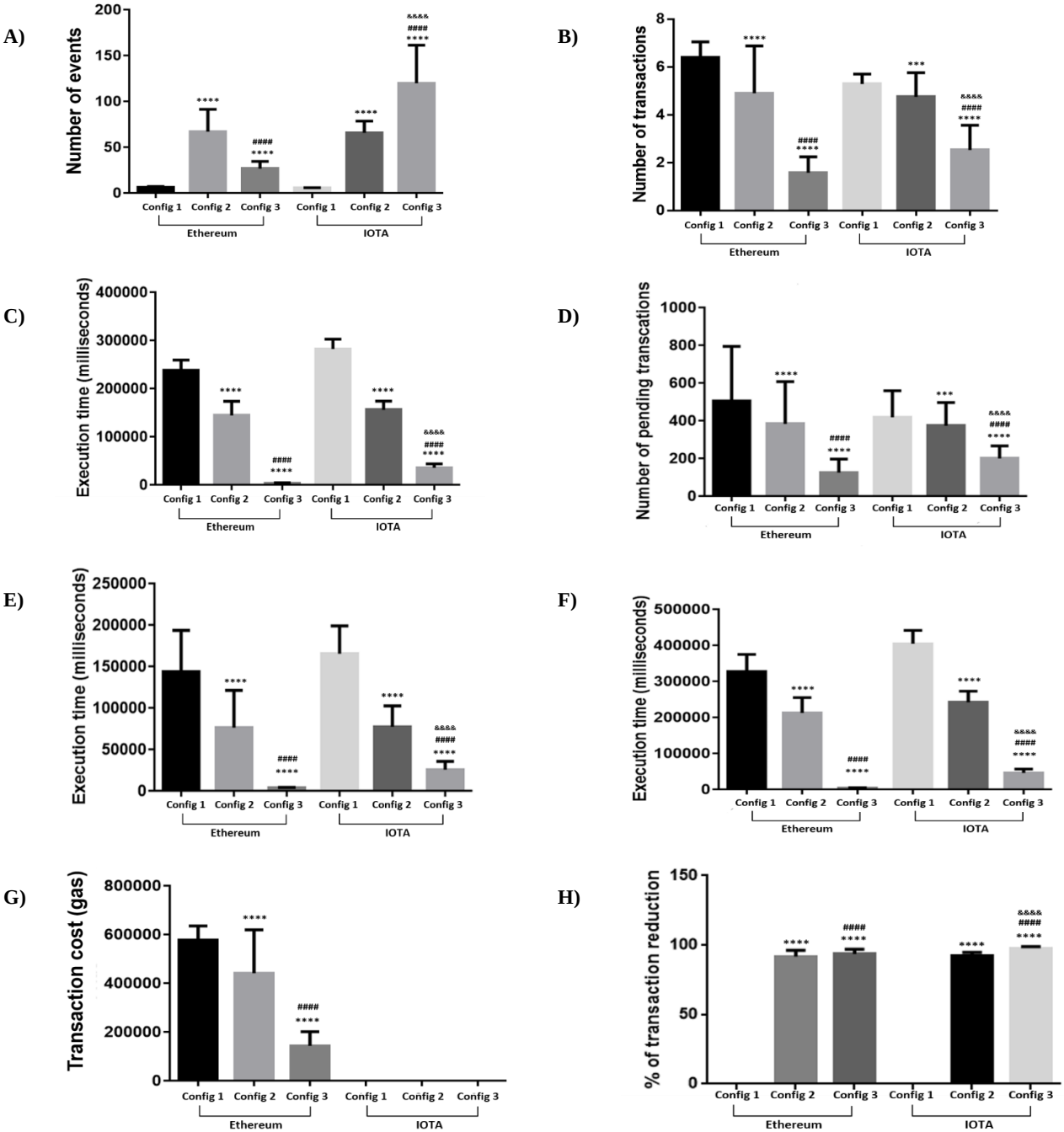

Figure 3. Graphical representation of the performance of Ethereum and IOTA network, evaluating six parameters: average of number of events **(A)** and transactions **(B)** per sensor; average running time **(C)**; number of pending transactions **(D)**; average of minimum **(E)** and maximum **(F)** execution time values; and transactions cost **(G)**. The percentage of transactions reduction was also plotted **(H)**. Three different configurations were analyzed: without data buffer and without CEP (*Config 1*); with data buffer and without CEP (*Config 2*); and with data buffer and CEP (*Config 3*), of both ledger evaluated. The data shown the averages and standard deviations of three independent experiments. Comparison among groups was performed by One-way ANOVA and $p < 0.05$ was considered statistically signifcant. Statistically significant differences between groups of the same ledger were represented as: **(*)** Analysis in relation to the *Configuration 1*; **(#)** Analysis in relation to *Configuration 2*. Comparisons between *Configuration 3* of Ethereum and IOTA were also carried out, where **(&)** represents statistically significant differences between both groups.

greatest efficiency in the reduction, with a percentage of 97.88% in relation to the control condition; being 3.84% higher than the reduction percentage obtained with the Ethereum ledger in the same condition (**Figure 3h**).

5. Discussion

To access the distributed ledgers through ContextNet, the DLedger service was implemented, which is responsible for connecting the IoT applications with the different immutable public records, and therefore, facilitates the storage of data in the ledger. With this service it is possible to choose between different technologies for data storage in distributed ledgers, maintaining the integrity and privacy of the data. One of the advantages of using ContextNet as access layer is the use of its functionalities that allow improving performance locally, at the edge of the network, using Fog computing. The use of this middleware also reduces the amount of data before they are sent to the cloud, by allowing the use CEP [Talavera 2016] and data buffer. CEP service filters events and only those that are relevant and necessary are stored; while data buffer accumulate the transactions.

To ensure data storage using distributed ledgers, two different technologies were used: the Ethereum blockchain and the IOTA tangle. Both technologies have the same algorithm (PoW) for the validation of transactions; although its use difficult to handle multiple IoT device requests, resulting in the loss of some data before its insertion into the ledger. To minimize these data losses, some of the main features of ContextNet were used to work with IoT devices.

The first blockchain integrated in the project was Ethereum, which has a large community of new developers, as well as being the first blockchain to implement smart contracts for autonomous applications. The Ethereum blockchain provides a persistent structure where data is stored in a reserved space in each transaction. This ledger has difficulty in partitioning and parallelizing the transactions [Wood 2018], so it cannot manipulate much information, which limits the scalability during the implementation of an IoT system. The use of data buffer with the Ethereum blockchain allowed the reduction of transactions in relation to the simple transactions due to the accumulation technique, which allows to reduce a set of events to a single transaction by concatenating the data of the events This led to a reduction in the average execution time and, as a result, the number of pending transactions and the total fee cost also decreased. In addition, the combination of data buffering and the CEP service further reduced the values of previously analyzed parameters in relation to the use of data buffer alone. These transaction reduction techniques, each separately, provided an increase in the performance of the network nodes, but the best results of the system were obtained with the combination of both of them.

In addition, the IOTA ledger was also added to the DLedger service. When analyzing the Tangle performance using the different filtering and accumulation techniques, a similar behavior to the Ethereum network was observed. As mentioned above, filtering queries made by CEP allow reducing the number of events that become transactions, while the data buffer concatenates numerous transactions, sending several at the same time. The decrease in the number of transactions improves the performance of the system, also decreasing the waiting time to process new transfers and the number of pending transactions.

Overall, in both distributed ledgers the best results were obtained through the combination of data buffer with CEP, where a statistically significant reduction in the values of all the parameters was observed. Thus, an improvement in system performance was observed. In addition to reducing the number of transactions, the use of both techniques also reduces the rate that must be paid for the validation of transactions. In the case of Ethereum, it would be the gas paid to the network by the mining company and there was a reduction of approximately 433949 units of gas by using CEP and data buffer, which represents a considerable saving. In the case of Tangle, only the energy consumed by the nodes is taken into account, since the cost of inserting the transaction is 0. This is an important parameter when deciding which ledger to use, since even with the reduction in this value obtained with the use of both accumulation techniques, the expense can be considerable, making the entire process more expensive.

When analyzing the reduction in the number of transactions it is possible to conclude that the IOTA network, in combination with data buffer and CEP, showed the better performance in relation with the Ethereum network in the same conditions. This is due to the process of parallelization in the validation of transactions in the Tangle network. However, the Ethereum network showed a greater reduction in the number of pending transactions and average execution time. This may be due to the IOTA network processing a greater number of events, which may have increased the processing time of the total transactions, and consequently, increase the waiting time.

Previous works, such as Cowry Platform [Odiete 2018], have used blockchain for the storage of IoT metadata, but this is the first time that transaction accumulation and filtering techniques are used to increase the efficiency of two different ledgers. Although we strongly recommend the use of the IOTA network using CEP and data buffer, due to the results obtained; we consider that multiple factors must be analyzed individually in order to select the ledger and the ideal configurations for each project.

In distributed ledgers, the validation of the network is an intrinsic property, so the adoption of peer-to-peer computing, such as the blockchain of Ethereum or Tangle of IOTA, to process and store all these transaction in the IoT scale, can also facilitate and reduce the costs associated with the installation and maintenance of large centralized data centers. The solution will be part of ContextNet as a new service for IoT gateways and the use of decentralize ledgers wills significantly increases the security of data storage with this middleware.

6. Related Work

The issue of privacy and the anonymity of data as a problem for IoT have been widely discussed; and several authors, such as Zyskind (2015), suggest the use of blockchain to solve it. Bitcoin [Nakamoto 2008] or Ethereum [Buterin 2015] distributed ledgers allow the storage of data in the blockchain, but with very little storage capability and at a very high cost.

Blockstack [Muneed 2016] and Metadisk [Wilkinson 2014] are two projects that use blockchain networks for data storage with proof of work algorithm, the same consensus algorithm of Ethereum or Bitcoin. Blockstack is an open source project where a name storage system, based on the blockchain of Namecoin, was created. The aim of Blockstack was to create a secure and human-readable network of names, to link these names with

some arbitrary values. To solve the problems related to the limitation of the storage capacity, some lists of links of named pairs of values were made. Initially, the project was using the Namecoin blockchain, which is an initial bitcoin fork; but in the end, the whole project migrate to the blockchain od Bitcoin due to security problems related to the vulnerability of the previous network, in relation to a critical security problem in which a single miner consistently had more than 51% of the total computing power.

Metadisk, on the other hand, is another open source data storage project created to demonstrate conceptually that blockchain data storage can be more decentralized, secure and efficient. They propose an autonomous and trustful cloud storage model in which users can upload and download files from the network in a secure manner. The blockchain is used as a data store only for file metadata, and the cryptocurrency is used as a payment mechanism to exchange storage space and bandwidth. Both projects has scalability limitations because of the block structure and the consensus algorithm [Wood 2014] which makes it difficult to reuse it in IoT systems. In addition, in order to insert data in the blockchain that exceed the space limit of the transaction, extra non-relational databases were used, which solve the storage difficulties of these networks.

In addition, new ledgers have emerged addressed to the storage of data, such as Filecoin, a blockchain which stores data in the blockchain within the transactions, using of proof-of-storage algorithms. This proof-of-storage algorithm is a variant to the PoW algorithm.

Although the blockchain is a relatively new technology, its use in the Internet of Things is increasing, mainly in data storage [Samaniego and Deters 2016], the security of devices with limited capabilities and micropayment; however, there are still challenges that must be addressed with respect to scalability and efficiency [Khan and Salah 2018].

7.Conclusions

We have developed a secure transaction storage system using distributed ledgers as a middleware service for the IoT Middleware ContextNet. The use of ledger for the storage of context data of IoT devices is an interesting solution, due to its decentralized, trustful and distributed architecture. With the proliferation in blockchain technologies, the number of available ledgers has increased, as well as the optimization mechanisms to improve the quality of the service. Factors such as the reduction of the number of transactions, the execution time, the number of pending transactions and the cost of the process must be taken into account, having different relevance depending on the aims of each project. After evaluating the performance of each ledger using accumulation and filtering techniques, we observed that the use of CEP and data buffer significantly increased the performances in both, the Ethereum and IOTA networks, in relation to the absence of these techniques or the use of data buffer alone; which was reflected in a reduction of the frequency and number of transactions transmitted.

The results presented in this paper demonstrated the effectiveness of the use of accumulation and filtering techniques in the Fog to improve the performance of distributed ledger networks. However, the IOTA network, which showed better performance, has not yet implemented smart contracts services. These smart contracts are very useful for the development of autonomous IoT applications. An alternative is to use the DLedger service

of ContextNet, which allows combining the best characteristics of the two analyzed ledgers. The IOTA network facilitates a DAG that solves scalability problems with the parallelization of validations; and with the DLedger service, a layer of the smart contracts offered by the Ethereum network over the IOTA tangle can easily be created. It would be interesting if future research implements this alternative in order to obtain a more complete system, so that it can be used by autonomous applications for the Internet of Things. We also propose to explore new options for the reduction of transactions to the network with the compression of the data, as well as to incorporate new technologies of different distributed ledgers to the ContextNet middleware.

Acknowledgements

The ContextNet project, and this work specifically, are partially supported by the FairComand and CAPES(Adrian´s Scholarschip) and CNPq grants Edital UNIVERSAL 2018 (Nr.:433183/2018-7)and DTI (Proc.304579/2017-3).

References

Bahga, A and Madisetti, V. K. (2016) "Blockchain Platform for Industrial". In *Journal of Software Engineering and Applications.* 9(1): 553-546. ISSN Online 1945-3124

Buterin, V. (2015) "A next generation smart contract & decentralized application platform". Ethereum White Paper.

Christidis, K. and Devetsikiotis, M. (2016) "Blockchains and Smart Contracts for the Internet of Things". IEEE Access, 2016, 4(1): 2169-3536. DOI 10.1109/ACCESS.2016.2566339.

Haller, S. (2010) "The Things in the Internet of Things". Tokyo, Japan: Internet of Things Conference, 2010. http://www.iot2010.org/.

Khan, M. A., and Salah, K. (2018). "IoT security: Review, blockchain solutions, and open challenges". Future Generation Computer Systems.

Muneed, A.; Nelson, J.; Shea, R. and Freedman M. J. (2016) "Blockstack: A Global Naming and Storage System Secured by Blockchains". USENIX Annual Technical Conference.

Nakamoto, S. (2008) "Bitcoin: A peer-to-peer electronic cash system". Bitcoin White Paper.

Odiete, O., Lomotey, R. K. and Deters, R. (2018) "Using Blockchain to support data and service managment in IoV/IoT". AISC Springer International, Saskatchewan. 733: 344-362.

Popov, S. (2016) The tangle. p. 131.

Razzaq, M. A, Gill, S. H., Qureshi, M. A. and Ullah, S. (2017) "Security Issues in the Internet of Things (IoT): A Comprehensive Study". *In International Journal of Advanced Computer Science and Applications.* p 6.

Robins, D. (2010) "Complex event processing". Second International Workshop on Education Technology and Computer Science. Wuhan, China.

Samaniego, M. and Deters, R. (2016) “Blockchain as a Service for IoT”. IEE International Conference on iThings, GreenCom, CPSCom and SmartData. Chengdu, China.

Endler, M. and Silva F. (2018) “Past, Present and Future of the IoMT Middleware”. Open Journal of Internet of Things(OJIOT), vol 4, nr. 1, pages 7-23, ISSN=23647108, July 2018.

Talavera, L. E., Endler, M. and Colcher, S (2016) “An Energy-aware IoT Gateway, with Continuous Processing of Sensor Data”. XXXIV Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos- SBRC 2016. Salvador de Bahia, Brazil.

Talavera, L. E., Endler, M., Vasconcelos, I., Vasconcelos, R., Cunha, M. and Silva, F. (2015) “The Mobile Hub Concept: Enabling Applications for the Internet of mobile Things”. 12th IEEE Workshop on Managing Ubiquitous Communications and Services (MUCS 2015), IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops), pp. 123-128, March 2015.

Vitalik, B. (2015) “A next generation smart contract & decentralized application platform”. s.l.: Ethereum White Paper.

Wilkinson, S. L. (2014). “Metadisk: Blockchain-Based Decentralized File Storage Application”. Technical Report.

Wood, G. (2014) “Ethereum: a secure decentralised generalised transaction ledger”. Ethereum Project Yellow Paper. 151: 1-32.

Xia, F., Yang, L. T., Wang L. and Vinel, A. (2012) “Internet of Things”. In *International Journal of Communication Systems* (Int. J. Commun. Syst.). 25:1101–1102 .

Zyskind, G. (2015) “Decentralizing Privacy: Using Blockchain to Protect Personal Data”. Security and Privacy Workshops (SPW), IEEE. IEEE.