



Project Report

Performance Evaluation of Broadcast Domain on the Lightweight Multi-Fog Blockchain Platform for a LoRa-Based Internet of Things Network

Muhammad Yanuar Ary Saputro  and Riri Fitri Sari * 

Department of Electrical Engineering, University of Indonesia, Depok 16424, Indonesia; m.yanuar71@ui.ac.id
* Correspondence: riri@ui.ac.id

Abstract: The Internet of Things (IoT) is a technology that allows every object or item to become part of the Internet and interact with each other. One of the technologies based on the IoT is Long Range (LoRa). Apart from the increasing number of IoT services, security aspects become a separate issue in the development of the IoT. One of the solutions is to utilize blockchain technology in the IoT topology to secure the data and transactions that occur in the IoT network. The blockchain can take minutes to compute a cryptographic chain. It also needs sufficient computing resources. This problem gave rise to the idea of establishing a lightweight blockchain platform with low latency that could run on devices with low computing resources as well as IoT devices. We offered a technology called Lightweight Multi-Fog (LMF) in our previous publication that is implemented using the Lightweight Scalable Blockchain (LSB) algorithm and the fog network on the IoT to solve the problem of integrating a blockchain with the IoT. In this paper, we simulate how the broadcast domain on LMF works and verify the results in lower latency and energy transmission compared to the standard blockchain model. The results showed that the average increase of the total delivery time ($T_{average}$) on the LMF platform was smaller than the average increase of the total delivery time ($T_{average}$), which was 0.53% for the variations in the number of nodes and 0.27% for the variations in the number of brokers/miners. Regarding the average increase of the total energy delivery ($E_{average}$), the Proof of Work (PoW) platform has a smaller increase of the total energy delivery ($E_{average}$), which is 1.68% during the variations in the number of nodes. In contrast, the LMF platform has a smaller average increase of the total shipping energy ($E_{average}$), which is 0.28% for the variations in the number of brokers/miners.



Citation: Saputro, M.Y.A.; Sari, R.F. Performance Evaluation of Broadcast Domain on the Lightweight Multi-Fog Blockchain Platform for a LoRa-Based Internet of Things Network. *Energies* **2021**, *14*, 2265. <https://doi.org/10.3390/en14082265>

Academic Editor: Daniela Mazza

Received: 23 February 2021

Accepted: 13 April 2021

Published: 17 April 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: IoT; LoRa; blockchain; latency; LMF; LSB; lightweight

1. Introduction

The Internet of Things (IoT) is a popular new concept in wireless communication technology. The basic idea of this concept is that various things or objects close to us, such as cameras, sensors, cards, Radio Frequency Identification (RFID), control arrays, phones and tablets, can communicate through integrated networks to carry out certain functions [1]. The most amazing thing about the IoT is not only limited to industrial automation but also related to how we live. For example, a smart home will allow people to turn on the lights, water and air conditioners as soon as they get home [2]. The IoT itself is sometimes referred to as Machine-to-Machine (M2M) technology [3]. The IoT is slightly different from M2M because it is a machine that communicates not only with other machines but also with sensors and humans [2].

Viewed from a general network perspective, the architecture used by the IoT can be categorized into three basic networks, namely, point-to-point, star, and mesh [4], as shown in Figure 1.

The extent of the applications of the IoT and types of IoT devices still face one obstacle, namely, security. Security is one of the most common issues in current IoT networks. The

most exciting case example is the Mirai botnet in September 2016, which disclosed a serious vulnerability in Internet of Things (IoT) devices. Originating from a blog as a target, the attack has become the highest Distributed Denial of Service (DDoS) attack until then. This attack was carried out by putting a pair of 62 usernames and passwords that generally exist on IoT devices and then turning them into botnets, which are then used to carry out DDoS attacks on certain web pages and services. The Krebs on Security blog was hacked using a DDoS attack with Mirai and BASHLITE on 20 September 2016. Additionally, Ars Technica reported an attack on the French site OVH [5]. In the future of 5G, IoT will play an important function, so security issues in IoT need to be handled quickly, especially for data protection. The boom in IoT has benefited many people and companies but is considered the most vulnerable point in cyber-attacks [6]. When IoT devices are being attacked, hackers will gain control and can steal personal sensitive data. Smart homes, for example, collect some personal and sensitive information. After hackers get accessed to it, they can know the user's behavior and preferences and use it for illegal activities [7]. Personal and sensitive information Collected on IoT devices can cause privacy concerns. Perhaps data could be protected in a centralized way to prevent this, but that would raise other concerns, such as a surveillance program. That is why data privacy in IoT needs to be protected, and one way to overcome this is by using a blockchain [7].

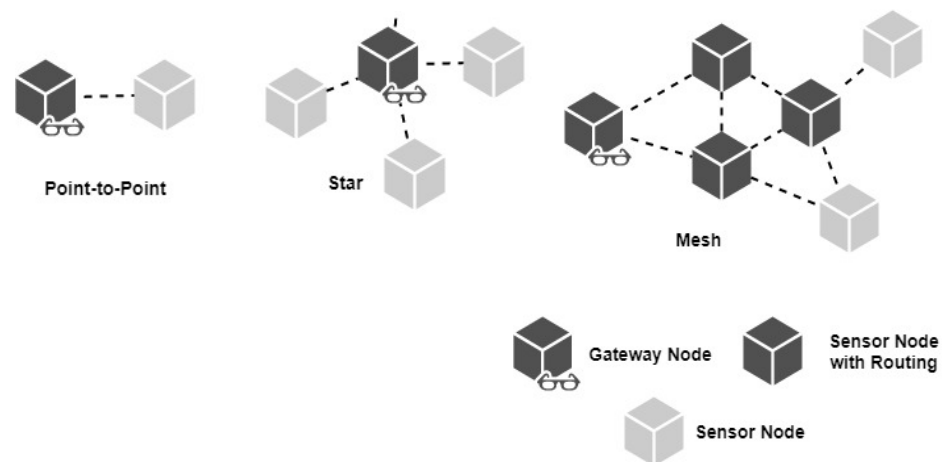


Figure 1. Network topologies commonly used by the Internet of Things (IoT) [4].

Blockchain technology has been studied by many researchers in various fields after its application was considered successful in the financial sector and smart contracts [8]. The success of the blockchain and its similarity to the IoT in terms of decentralization has led some researchers to try to implement the blockchain as a security system on the IoT. During the development of the IoT, the blockchain can make notes that are not easily changed and accessed by irresponsible parties.

Here are some advantages of a blockchain, which can improve the security of IoT devices, especially regarding data privacy [8]:

1. Decentralized. The decentralized IoT architecture with blockchain uses a blockchain as a solution that has high scalability. It is also resistant to DDoS attacks and Single Point of Failure (SPOF) problems.
2. Pseudonym. A node in a blockchain is identified based on its public key or hash, so it does not provide information about the participating nodes.
3. Transaction security. Each deal should be signed, verified and validated by the miner or broker prior to being forwarded to a blockchain network. Once validated, it will be difficult to change because it has been stored on a blockchain.

Transaction security on a blockchain produces transparency and any modification can be easily tracked and detected; thus, a blockchain is able to improve IoT security, especially on a privacy level. Users also will not be worried about any surveillance program launch

by the government or company. By using a blockchain, the data are not centralized, and this will solve the IoT privacy issue [7].

Unfortunately, the implementation of the blockchain as a security system on the IoT network was not as smooth as had been imagined. Some researchers tried to implement a blockchain on an IoT network. Among the various efforts is IoTChain, which uses a blockchain as an authentication, authorization and recording mechanism (Authentication, Authorization and Accounting/AAA) [9]. Another platform is BeeKeeper, an Ethereum-based blockchain implementation [10]. Both studies resulted in the successful application of the blockchain to the IoT. However, both IoTChain and BeeKeeper's main problems are in the integration of the blockchain and the IoT, namely, the high computational resources and latency for processing each transaction.

For Bitcoin, it could take 30 min to verify a transaction. Furthermore, existing devices or IoT nodes generally have limited hardware capabilities [11]. The large computational resources and latency required by the blockchain make blockchain implementation in an IoT network difficult. Therefore, researchers have also tried to develop new mechanisms in addition to Proof of Work (PoW) or Proof of Stake (PoS), so that the blockchain could be applied to the IoT network. Some solutions that have emerged are the application of fog and cloud computing on the IoT network, which is called FogBus [12], and the Lightweight Scalable Blockchain (LSB) system [13]. This can reduce the latency and computing resources needed.

All models have their strengths and weaknesses, such as the level of security and privacy offered by LSB and the flexibility and scalability offered by FogBus. However, LSB and the blockchain in general still use a broadcast domain. All brokers are assumed to be on the same broadcast domain so that when there is an attack on one broker, it will be easier for the other brokers to be attacked. In addition, LSB has a mechanism to limit the number of transactions when there is an attack. This reduces the level of availability and disrupts communication and processing. However, on FogBus, there have not been any tests or a focused analysis on its security [11]. The latency obtained is still quite high when using blockchain technology.

This gave rise to the idea of making a lightweight blockchain model with low latency that could run on devices with limited computing capabilities, such as IoT devices. In our previous publication, we proposed a technology called Lightweight Multi-Fog (LMF), using the ability of the Lightweight Scalable Blockchain (LSB) algorithm and the fog network on the IoT to solve the problem of integrating the blockchain on the IoT to increase the IoT security [14]. Lightweight Multi-Fog (LMF) is designed with the intention of being used in large-scale areas consisting of several cities or regions, which is represented by Broadcast Domains [14]. Long-ranged Wireless Sensor Networks will be perfect as a simulation scenario for LMF.

There are many wireless sensor networks that are used to implement the IoT, such as ZigBee, LoRa, Sigfox and NB-IoT. LoRa, Sigfox and NB-IoT have a long range compared to ZigBee, which is why they can be adapted to scenarios that need long-range transmission [15]. Scenarios such as agriculture, smart farming, smart cities and their combination require wide-area transmission, so a wide-range Wireless Sensor Network is needed in these scenarios. NB-IoT uses licensed frequency, unlike Sigfox and LoRa, which use unlicensed frequency [15]. LoRa and Sigfox are also more mature and are the most used globally [15,16]. Although NB-IoT has the highest data rates compared to the other two, it relies on LTE technology coverage, which is not available in remote or rural areas in developing countries [17]. NB-IoT may be able to offer better Quality of Services (QoS), latency and scalability than Sigfox and LoRa, but its power consumption is the highest of all [15].

Sigfox does not support authentication or encryption and also does not support private networks, as opposed to NB-IoT and LoRa, even though it has a wider range [15]. Compared to Sigfox and other networks, LoRa is the best choice in flexibility if we want to use it on public or private networks and operator-based or private deployment models [15,16]

with long-ranged coverage, high data rate and low power consumption. LoRa has three Class Option, namely, Class A, Class B and Class C. LoRa also has customizable parameters to adjust the environment [16]. As a comparison of all Wireless Sensor Networks, LoRa is used as an LMF simulation scenario in this paper, which has several similar characteristics to LMF. LMF and LoRa can both be used in large-scale areas, in private or public networks, have small latency with low power consumption and are customizable.

In this paper, we simulate how the broadcast domain on LMF works and verify the results regarding the latency and energy transmission compared to those of the standard blockchain model. The contribution of this article is to bring the detailed Broadcast Domain fault-tolerance design, create a simulator that simulates how the Broadcast Domain of LMF works on a LoRa network named LMFSim, conduct a performance evaluation of the Broadcast Domain design using the LMFSim simulator, compare it with PoW and provide an analysis of the Broadcast Domain's performance based on the energy consumption and processing time. LoRa is used as a simulation scenario because it has similarities with the characteristics of LMF and the intention of LMF.

This paper consists of four further sections. Section 2 explains how the broadcast domain on LMF works and how the simulator works. Section 3 explains the simulation scenario and the algorithm for the broadcast domain simulation. Section 4 explains the results of the simulation and the paper is concluded in Section 5.

2. Lightweight Multi-Fog (LMF) Blockchain and LMFSim

The Lightweight Multi-Fog (LMF) layer, compared to FogBus [12] and LSB [13], is very distinct. LMF uses a functioned based layer to categorize each of its segments. First is the access layer, then the network layer and the computational layer and last is the application layer. The four layers of LMF are portrayed in Figure 2 [14].

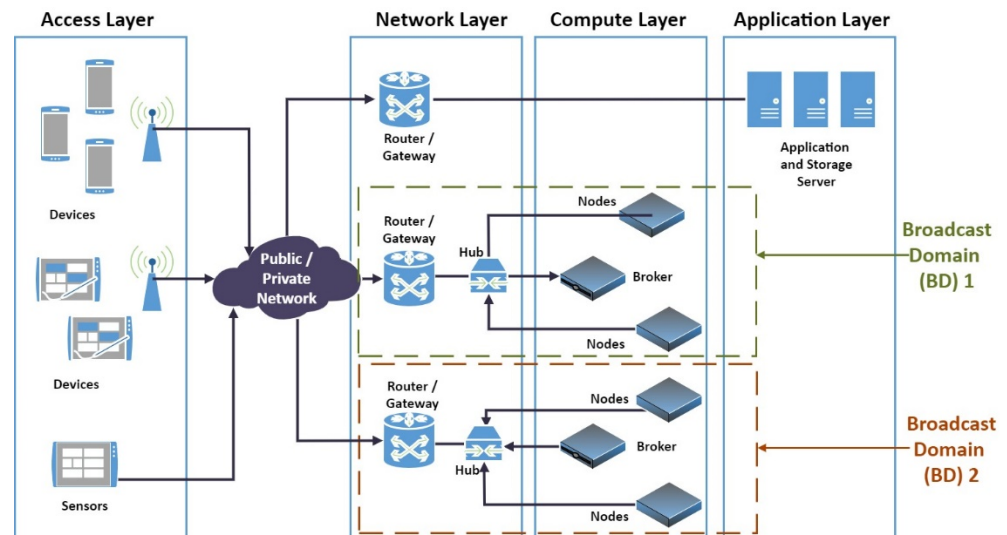


Figure 2. Lightweight Multi-Fog Topology (LMF) [13].

The lowest layer, which consists of IoT devices and sensors, is the access layer. This layer can be connected to the public Internet or a private local network. Therefore, the implementation of LMF could be on public or private networks.

The layer that performs network functions contains routers and switches are the network layer. Network devices in the network layer served as a gate. It sends data to the blockchain brokers and blockchain nodes. The network layer provides one gateway for each broadcast domain. The network layer has at least one routing network devices for each broadcast domain, which accommodates one broker and several nodes. Each broadcast domain can also be symbolized as a town, territorial, region or state. Therefore,

the number of broadcast domains is equivalent to the number of towns, territorials, regions or states where the technologies is applied.

The computational resources populate the computational layer, which has one broker and a certain number of blockchain nodes in each broadcast domain. Originally, the node simply handles the transactions from the broker on the same broadcast domain. If the broker is not available, then the node that has certain resources in the same broadcast domain will become a new broker. If there are no existent nodes with certain computational resources in a specific broadcast domain, the broker in the other broadcast domain will become the new broker for the specific broadcast domain.

The top layer in LMF is the application layer, which contains several application servers and database servers. The backup nodes also reside in this layer. The application servers stored and processed transactions and data in this layer.

2.1. Broadcast Domain on LMF

The Broadcast Domain (BD) is an area or group that contains nodes that will reply to all broadcast packets from every node within the same region or group. The Broadcast Domain in the LMF is applied to isolate each region. The regions can represent territorials, towns or states. This platform is adopted to mitigate large-scale DDoS attacks on node brokers. After a DDoS attack has occurred against one or multiple brokers on one broadcast domain, the attack will not influence the brokers on other broadcast domains [14].

Every broadcast domain has a node that acts as a broker. Another node act as a computational layer. None of these nodes communicate with distinct broadcast domain nodes, except when the broker goes down, but no node on a specific broadcast domain can serve as a broker [14].

The data mechanism on LMF is similar to LSB. The whole nodes in the broadcast domain have their independent Public Keys (PK). Each node would issue a unique public key for each transaction. Each block included the applicant's public key hash, the destination's public key hash for this specific transaction and the applicant's public key hash for the upcoming transaction [13]. This mechanism ensures that subsequent transactions are legal. This is done by comparing the applicant's public keys in subsequent transactions with the applicant's public keys that have been saved in previous transactions. Brokers also communicate with all the network's other brokers on a distinct broadcast domain. This communication authorizes transactions using indirect and direct evidentiary mechanisms [13]. This mechanism will lessen the time for verification.

Distant from LSB, LMF saves blocks in the local nodes on the broadcast domain. Arriving transactions to the broadcast domain will not be kept in the distinct node on a distinct broadcast domain. Each broadcast domain could possess a distinct blockchain because each broadcast domain has its distinct blockchain and its backup nodes in the cloud. Therefore, when all nodes on the broadcast domain are not available, the block is still kept in the backup node on the cloud. There was a distinction between cloud and local nodes, i.e., each node could have only one broker in a specific broadcast domain. Yet, the cloud node is a member for every broker on all broadcast domains because it acts as a backup node.

The status, availability, capacity of All nodes, brokers and cloud nodes are checked regularly by the Central Monitoring and Provisioning Application on the Cloud/Application Layer. This Monitoring and Provisioning Application is also the one that pushes the configuration files on all nodes, brokers and cloud nodes. When a broker in the specific broadcast domain is not available, the Application will push a new configuration to the nodes containing new Broker information. This mechanism is presented in Figure 3.

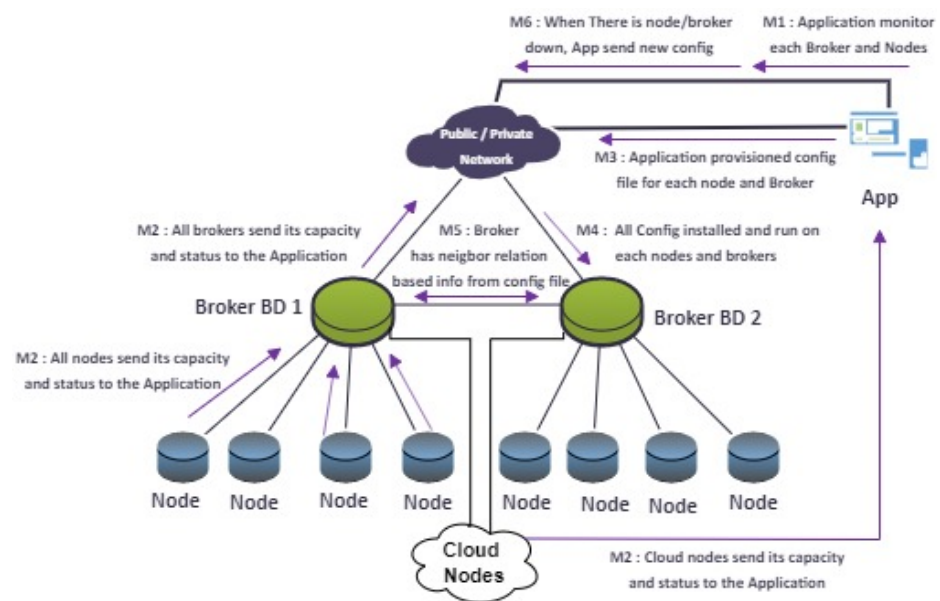


Figure 3. Broadcast domain communication mechanism on lightweight multi-fog topology.

The Application Server serves as a Monitoring tool, which monitors the availability status and capacity status on each broker and each node on all Broadcast Domains (M1). The application server also serves as a Provisioning tool, which pushes configuration files to each broker and each node on all Broadcast Domain (M3). The configuration file tells the node about its broker and Broadcast Domain information. When a broker in the specific broadcast domain is not available, the Application will push a new configuration to the nodes containing new broker information (M6). If the new broker is on a different Broadcast Domain, then the new configuration will include the new Broadcast Domain for the nodes. All nodes and brokers also send information about their status to the application when there is an error or low capacity (M2). As for the broker, the configuration file will also include the neighbor relationship between a broker in different Broadcast Domains (M5).

By default, brokers in different Broadcast Domains cannot communicate with each other without this neighbor table information. Neighbor table information is used by the broker to communicate with each other for authorizing the transactions using indirect and direct evidentiary mechanisms. Other than broker-to-broker communication for authorizing the transactions, no other transaction between a node to other node on different Broadcast Domain. As for Cloud Nodes, Cloud Nodes also have their independent Public Key (PK), similar to local nodes. Each transaction block will include the applicant's public key hash, the destination's public key hash for this specific transaction and the applicant's public key hash for the upcoming transaction [13]. The broker will verify this transaction using direct or indirect evidence to another broker. When there is no evidence before, the broker will verify and validate it first and then broadcast this transaction, which will be signed by all nodes in a specific Broadcast Domain. However, if there is evidence before, this transaction has already been verified, and the broker will not need to validate it again.

The Broadcast Domain on the Lightweight Multi-Fog blockchain (LMF) has its ledger or chain separate from each other, except for Cloud nodes. Normally, only brokers on the same Broadcast Domain could be accessed. It stores data on the node with the same Broadcast domain. However, if there are attacks on one Broadcast Domain, other brokers on different Broadcast Domains can become brokers of a failed Broadcast Domain, with Cloud nodes as its node.

Figure 4 shows the flow when the broker is down or has failed. Other nodes can become the new broker for the failed Broadcast Domain; for example, Broadcast Domain 1 (BD1). As long as there is an available node on the failed Broadcast Domain that can become a new broker, no broker is selected outside of the failed Broadcast Domain. The

application selects the node with the highest available compute resources and the lowest latency to the Applications and the other brokers.

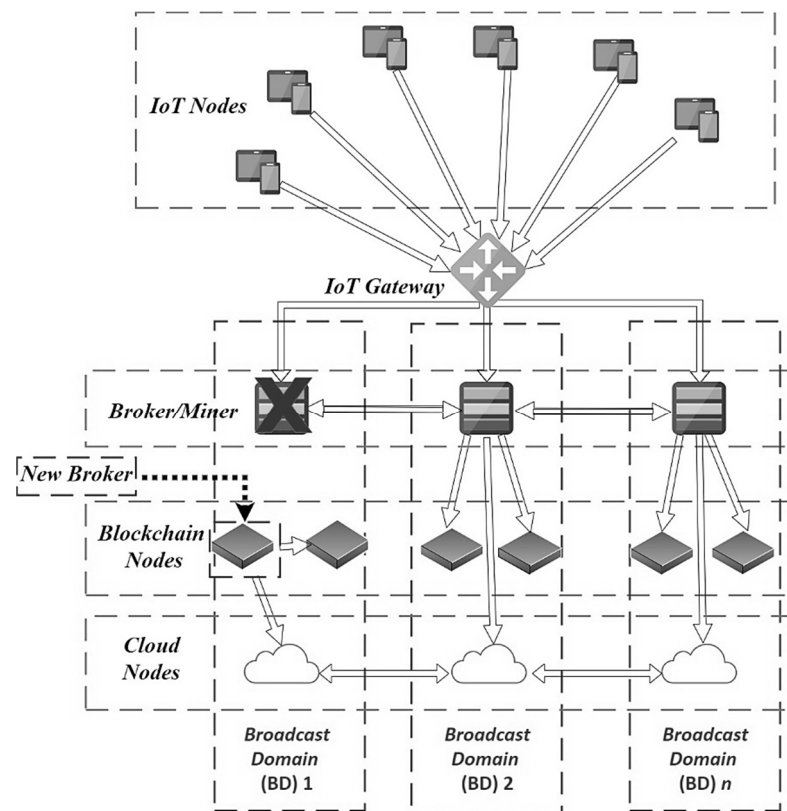


Figure 4. The broker on BD 1 failed, and thus, one node becomes the new broker.

The Broker outside can be selected if all nodes on the failed Broadcast Domain are down or when there are attacks on a failed Broadcast Domain. A Failed Broadcast Domain is discovered by the Application when the Application is unable to communicate with the broker and all nodes in a particular Broadcast Domain. The application will mark the Broadcast Domain as failed if it cannot communicate with the broker and all nodes in it. The new Broker will have the blockchain for the failed Broadcast Domain because it has Cloud Nodes which are members of all Broadcast Domains. Figure 5 shows that the Broker on Broadcast Domain 2 (BD2) becomes the new broker for Broadcast Domain 1 (BD1), which failed due to attacks. The broker from Broadcast Domain 2 (BD2) becomes the new broker and will have the database from Broadcast Domain 1 (BD1), because it has Cloud nodes that are a member of all Broadcast Domains. The Cloud Node on Broadcast Domain 1 (BD1) and Broadcast Domain 2 (BD2) are one entity with two or more nodes in the same cluster. The Application will select a broker from Broadcast Domain nearest to the IoT nodes or users to become the new broker for the failed Broadcast Domain.

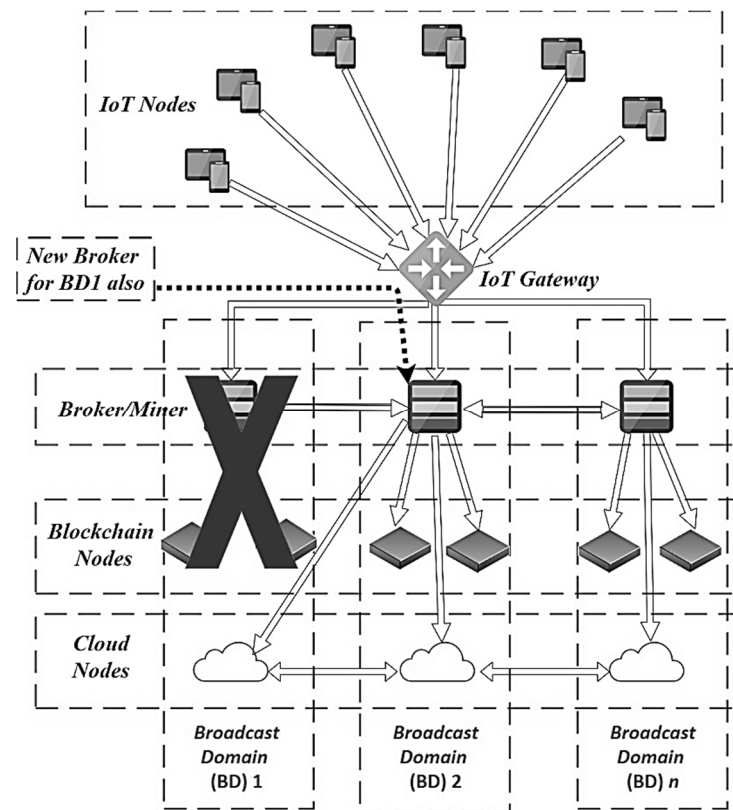


Figure 5. The broker on BD2 becomes the New Broker for BD1 if all nodes failed on BD1.

2.2. LMF's Broadcast Domain LMF Simulation Using LMFSIM

In these papers, we compared the performance of the Lightweight Multi-Fog blockchain (LMF) topology model, where the broadcast domain is separated based on the location with the blockchain topology model of the Proof of Work (PoW) method on the LoRa network. In this research, a simulation of the IoT LMF network broadcast domain is made using the Simpy 3.0.11 program instrument. The program also simulated packet transmissions that pass through the network with several variables that varied to discover the transmission time needed and the amount of energy expended by the node.

This study uses LoRa to simulate sending a packet from a random node to the Base Station (BS), after which the packet is then sent to the broker for all nodes in the PoW method or in a Broadcast Domain (BD) in the LMF method. The broker will then conduct a mining process, which consists of identification, authentication, authorization and verification, to then be stored in a block and sent to all nodes that are on the same broadcast domain in the form of a blockchain.

Based on the modeling, the parameters to be measured are as follows:

- The total transmission time ($\sum T$) is the amount of time needed to send packets from the original node to the BS ($\sum T_{BS}$) and the amount of time needed to send the blockchain from the BS to all nodes in one BD ($\sum T_{BC}$):

$$\sum T = \sum_{i=0}^n T_{BS} + \sum_{i=0}^n T_{BC} \quad (1)$$

- The total amount of energy ($\sum E$) is the amount of energy needed to send packets from the original node to the BS ($\sum E_{BS}$) and the amount of energy needed to send the blockchain from the BS to all nodes in one BD ($\sum E_{BC}$):

$$\sum E = \sum_{i=0}^n E_{BS} + \sum_{i=0}^n E_{BC} \quad (2)$$

The LMFSim simulator (<https://github.com/myanuararys/lmfsim>, accessed on 10 March 2020) was used here, which is a combination of the LoRaSim simulator [18] and the BlockSIM simulator [19], which runs on the Linux operating system Xubuntu 18.04 LTS. LMFSim has not yet simulated full consensus on LMF of Broadcast Domain failover mechanism. It only simulates packet transmission on LoRa with multiple nodes or Broadcast Domain with a simple PoW and LMF Broadcast Domain model. The parameter settings in the LoRa simulation use the SN5 set type. SNn defines a LoRa configuration parameter set where the Spreading Factor (SF), Transmission Power (TP), Bandwidth (BW) and Carrier Frequency (CF) are set the same in all experiments [18]. The number of packages sent at each trial was set to 20 bytes.

The simulation was run using 1 Base Station (BS) or gateway for all nodes on the LoRa network. A packet of 20 bytes is sent from one of the nodes chosen at random to the Base Station (BS). Then, the packet is sent from the Base Station (BS) to all broker nodes to be verified, validated and encrypted and then sent to all existing nodes (on PoW) or nodes that are in a Broadcast Domain (BD) with broker nodes (on LMF). The maximum latency between nodes (T_{max}) was set to 20 ms.

The simulation is performed by varying the number of nodes (N_{nodes}) and the number of broadcast domains (N_{BD}). The value of the Spreading Factor (SF), Transmission Power (TP), Bandwidth (BW) and Carrier Frequency (CF) on the SN5 set type are 12 MHz, 14 dBm, 125 MHz and 868 MHz, respectively. The number of broadcast domains (N_{BD}) on LMF and the number of miner nodes (N_{miner}) cannot be less than the number of nodes (N_{nodes}). The number of broker nodes (N_{Broker}) on LMF is the same as the number of broadcast domains (N_{BD}). This is because every broadcast domain has a requirement to have one node functioning as a broker (broker node).

Some libraries and programs are required before running LMFSim. The base program is python3, simpy, numpy, python-pip, network, pandas and matplotlib. All of these programs can be installed using the apt-get function on Xubuntu. The hardware and specifications of the computer used in this paper can be seen in Table 1.

Table 1. Hardware and software specifications of the computer used in this study.

Hardware	Specification
Processor	Intel® Core® i5-3360M CPU @ 2.8 GHz
Number of cores/threads	8/16
RAM	2 GB
Software	Specification
Operating System	Xubuntu 18.04 LTS
Terminal emulator	Console/Terminal

In the LMFSim simulator, there are two classes. The first is named myNode, which creates IoT nodes. The second is blocknodes, which defines the blockchain variables of compute nodes, such as miner and broker. This class also defines receiver functions to solve cryptographic puzzles (mining), broadcast transactions from broker to all nodes and accept transactions on each node. Apart from the main class, there are several functions. First is the node_generator function, which configures nodes according to consensus and the variables defined in the config file. There is also the trans_generator function, to perform transaction generation, and the monitor function to store generated blocks. Last is the transmit function, which sends packets from each node to the Base Station (BS). Example of simulation can be seen in Figure 6.

```

-----
1000, 86, Created, block, 99910,0
hash of block is d41d8cd98f00b204e9800998ecf8427e
1000, 260, Created, block, 99910,0
hash of block is d41d8cd98f00b204e9800998ecf8427e
1000, 268, Created, block, 99910,0
hash of block is d41d8cd98f00b204e9800998ecf8427e
1000, 345, Created, block, 99910,0
hash of block is d41d8cd98f00b204e9800998ecf8427e
1000, 417, Created, block, 99910,0
hash of block is d41d8cd98f00b204e9800998ecf8427e
1000, 452, Created, block, 99910,210
hash of block is 114e07ab2f79eb80459b8f3c72888041
1000, 472, Created, block, 99910,0
hash of block is d41d8cd98f00b204e9800998ecf8427e
nrCollisions 4492
14.144
energy (in J): 41.297084174
sent packets: 4583
collisions: 4492
received packets: 10
processed packets: 546
lost packets: 0
SLA: 98.0144010473 %
LMFn5.dat
72.400810957
-----
Simulation ended
Total Time taken 72:
myanuar@ubuntu:~/Downloads/LMFSim/srcs █

```

Figure 6. Simulation result using LMFSim.

3. Simulation Scenario

Before the transmission of the packet takes place, the number of nodes and number of BDs to be simulated are determined, and then the nodes are grouped according to their broadcast domain. In this simulation, the number of nodes is divided equally according to the number of available BDs, and then, one broker is chosen for each BD. The broker will send to the node only the blockchain that is in the same BD as the broker. The distribution and transmission flow are illustrated in Figure 7.

The flow and distribution of the packages and blocks in this system can be stated as an algorithm as a reference for the design of the simulator.

The simulation was run with the assumption that the average time needed for packet transmission ($T_{avg\text{send}}$) was 10 ms and the average time needed to conduct mining ($T_{avg\text{mining}}$), i.e., the validation, authentication and verification process, at the broker or miner node was also at 10 ms. The simulation process was carried out for a certain time (T_{sim}), namely, 110 ms in each experiment. Algorithm 1 is being used for node creation and Algorithm 2 is being used for the packet transmission.

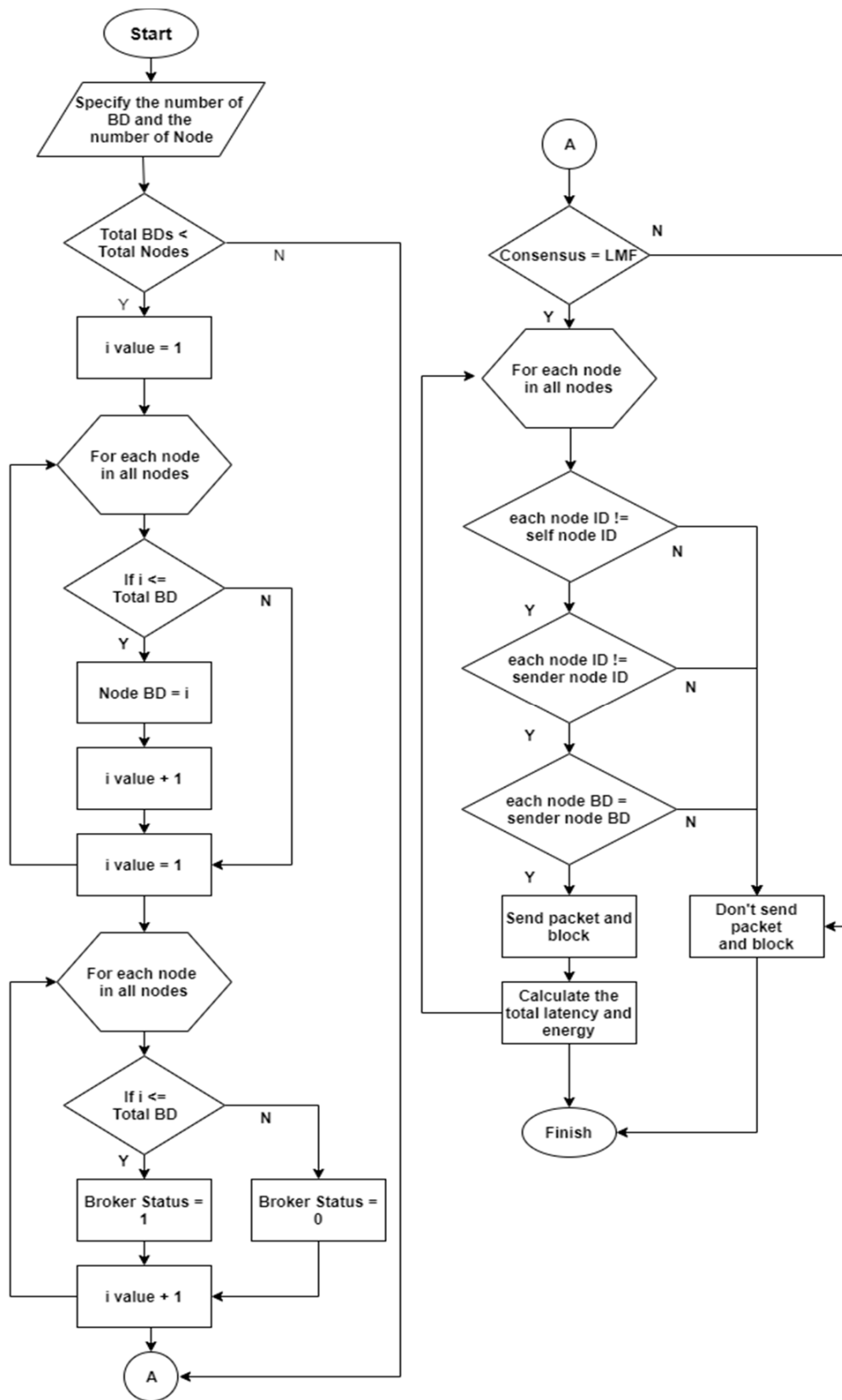


Figure 7. Flow distribution of Broadcast Domain (BD) and transmission on BD.

Algorithm 1. BD and broker allocation.**Input:** node list with class format, integer total broadcast domain and total node

```

1.  for each node:
2.      if total broadcast domains < total nodes:
3.          set i value to 1
4.          for each node in node list {
5.              if i <= total broadcast domain {
6.                  set node BD to i
7.              }
8.              else {
9.                  reset i value to 1
10.             }
11.             increasing i value by 1
12.         }
13.     set i value to 1
14.     for each node in node list {
15.         if i <= total broadcast domain {
16.             set broker status to 1
17.         }
18.         increase i by 1
19.     }
20. }
21. }

```

Algorithm 2. Packet Transmission.**Input:** string consensus, integer node ID, sender node ID, list node list, Boolean node BD**Output:** list packet, float latency

```

1.  if consensus is LMF {
2.      for each node in node list {
3.          if node ID != self ID and node ID != sender node ID and node BD == self BD {
4.              count latency between node
5.              transmit the packet
6.          }
7.          else {
8.              drop the packet
9.          }
10.     }
11. }

```

4. Results and Analysis

We calculated the total transmission time ($\sum T$) and total energy ($\sum E$) obtained from each experiment with different variables to measure the performance.

4.1. LoRa IoT Network Work System with the PoW Blockchain Model

The total transmission time ($\sum T$) is obtained from the first and second simulations. A comparison of the total transmission time and the number of nodes (N_{node}) can be seen in Figure 8. A comparison of the total transmission time value and the number of miner nodes (N_{Miner}) can be seen in Figure 9.

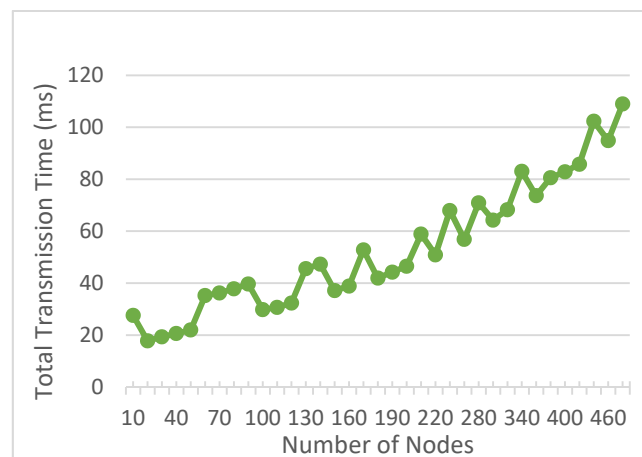


Figure 8. The total transmission time vs. the number of nodes for the PoW model.

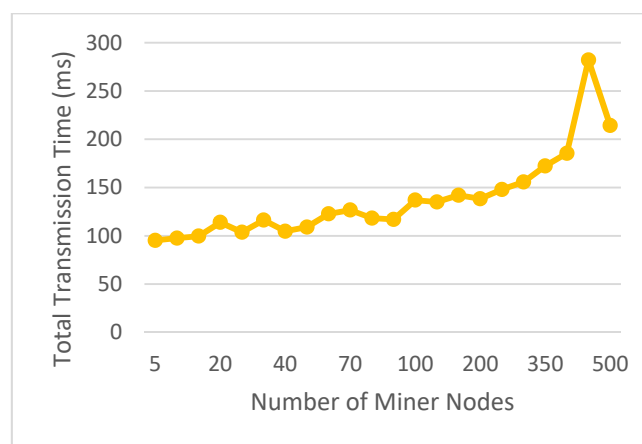


Figure 9. The total transmission time vs. the number of miner nodes in the PoW model.

Figure 8 shows that the greater the number of nodes (N_{node}), the higher the total transmission time ($\sum T$). It can be estimated that the time needed to process the mining and broadcasting of packages to all existing nodes increases with the number of nodes (N_{node}). This increase of the amount of time increases the total transmission time ($\sum T$).

Figure 9 shows that the greater the number of miner nodes (N_{miner}), the higher the total transmission time ($\sum T$). Increasing the number of miner nodes (N_{miner}) increases the number of nodes that conduct the mining. It also increases the time needed to send packets to all existing nodes. There is also a sudden increase of the total transmission time ($\sum T$) in Figure 9. This increase happens when the number of miner nodes (N_{miner}) reaches 450. This sudden increase is caused by the increase of the number of collision and the number of transmissions. The increase of the number of nodes increases the number of collisions and the number of transmissions. The distance between the nodes also becomes one of the aspects that can increase the total transmission time ($\sum T$). LMFSim used part of LoRaSim, which set the distance between the nodes randomly based on the Path Loss model [18].

It can be concluded here that on the IoT network with the PoW blockchain model, an increase of the number of nodes (N_{node}) or the number of miner nodes (N_{Miner}) will increase the total transmission time ($\sum T$), as shown in Figures 8 and 9. Equation (3) can be used to measure the average increase of the total transmission time ($T_{average}$) with respect to the number of nodes (N_{node}) or the number of miner nodes (N_{Miner}):

$$T_{average} = average \left(\sum_{i=1}^n \frac{\sum T_{i+1} - \sum T_i}{\sum T_i} / \frac{N_{i+1} - N_i}{N_i} \right) \times 100\% \dots \dots \dots (3)$$

Equation (3) shows that the average increase of the total transmission time per additional node is 0.72%. In addition, the average increase of the total transmission time per additional miner node is 0.9%. It can be concluded that increasing the number of miner nodes (N_{miner}) increases the amount of time needed for transmission more than when increasing the number of nodes (N_{node}).

We calculated the total amount of energy ($\sum E$) obtained from the first and second simulations. A comparison of the total energy used for sending with respect to the number of nodes (N_{node}) can be seen in Figure 10. In addition, a comparison of the total energy for transmission with respect to the number of miner nodes (N_{Miner}) can be seen in Figure 11.

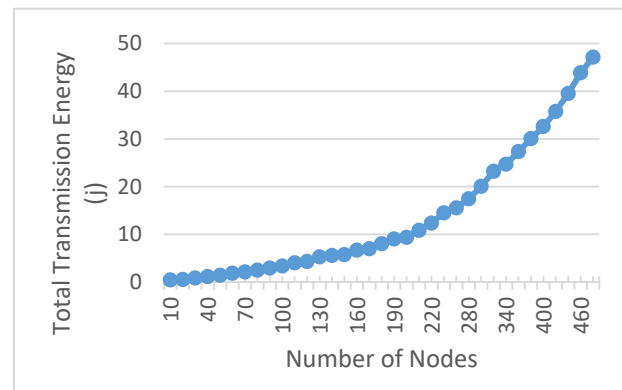


Figure 10. The total transmission energy vs. the number of nodes in the PoW model.

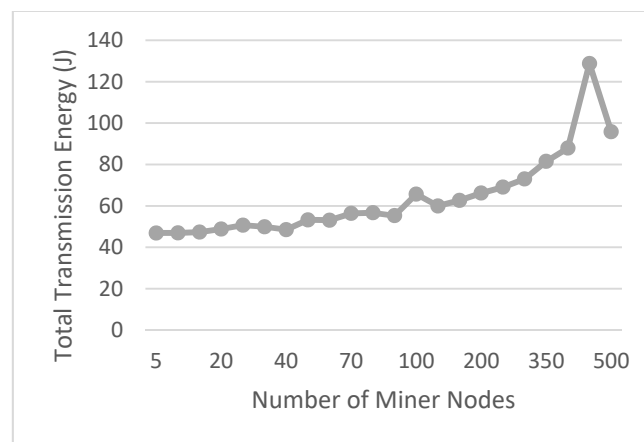


Figure 11. The total transmission energy vs. the number of miner nodes in the PoW model.

Figure 10 shows that the greater the number of nodes (N_{node}), the higher the total amount of energy ($\sum E$). It can be estimated that the energy needed to process the mining and broadcasting of packets to all existing nodes increases with the number of nodes (N_{node}). The increase of the amount of energy is what increases the total energy used for packet/block transmission ($\sum E$).

Figure 11 shows that the greater the number of miner nodes (N_{miner}), the higher the total transmission energy ($\sum E$). Increasing the number of miner nodes (N_{miner}) increases the number of nodes that conduct the mining, thus increasing the energy needed to send packets to all existing nodes. The sudden increase of the total transmission energy ($\sum E$) also happens in Figure 11. This increase happens when the number of miner nodes (N_{miner}) reaches 450. The reason of this sudden increase is the increase of the number of collisions and the number of transmissions. The increase of the number of nodes increases the number of collisions and the number of transmissions. The distance between the nodes is one of the aspects that can increase the total transmission energy ($\sum E$). LMFSim used part of LoRaSim, which set the distance between nodes randomly based on the Path Loss

model [18]. A highest number of collision and transmission happens when the number of miner nodes (N_{miner}) reaches 450.

It can be concluded that on the IoT network with the PoW blockchain model, an increase of the number of nodes (N_{node}) or the number of miner nodes (N_{Miner}) will increase the total transmission energy ($\sum E$), as shown in Figures 10 and 11. We use the following equation to measure the average increase of the total transmission energy ($E_{average}$) per node (N_{node}) or per miner node (N_{Miner}):

$$E_{average} = average \left(\sum_{i=1}^n \frac{\sum E_{i+1} - \sum E_i}{\sum E_i} / \frac{N_{i+1} - N_i}{N_i} \right) \times 100\% \dots \dots \quad (4)$$

Equation (4) shows that the average increase of the total transmission energy per node is 1.68%. In addition, the average increase of the total transmission time per additional miner node is 0.31%. This is in contrast with the increase of the total transmission time. Therefore, increasing the number of miner nodes (N_{Miner}) results in a lower increase of the energy needed for transmission than when increasing the number of nodes (N_{node}).

4.2. LoRa IoT Network Work System with LMF Blockchain Model

A comparison of the total transmission time to the number of nodes (N_{node}) can be seen in Figure 12. In addition, a comparison of the total transmission time versus the number of broadcast domains (N_{Broker}) can be seen in Figure 13.

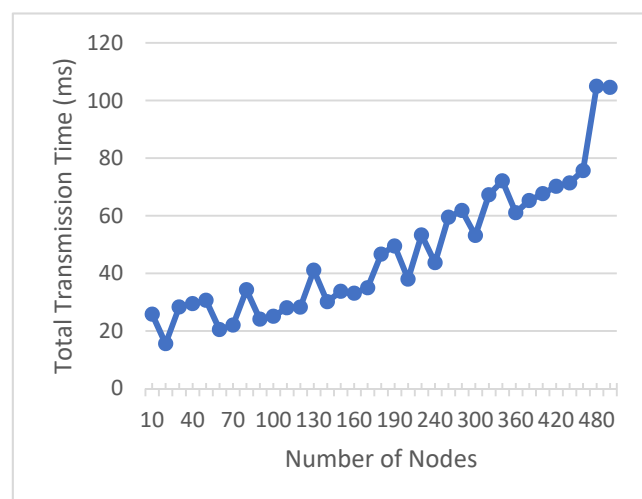


Figure 12. The total transmission time vs. the number of nodes in the LMF model.

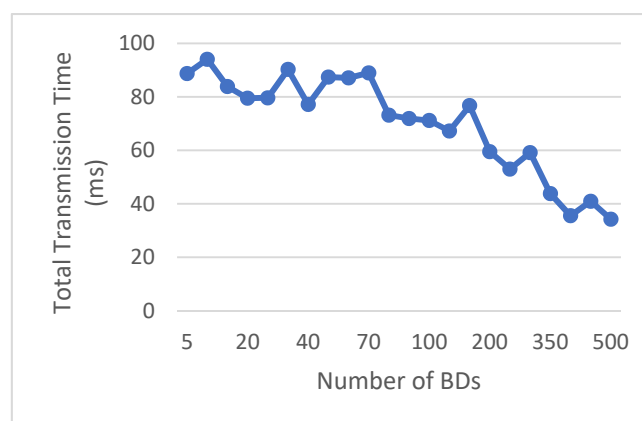


Figure 13. The total transmission time vs. the number of BDs in the LMF model.

Figure 12 shows that the greater the number of nodes (N_{node}), the higher the total transmission time ($\sum T$). It can be estimated that the time needed to process the mining and broadcasting of packages to all existing nodes increases with the number of nodes (N_{node}). This increase of the amount of time increases the total transmission time ($\sum T$). There is also a sudden increase of the total transmission time ($\sum T$) in Figure 12. This increase happens when the number of nodes (N_{node}) reaches 480. The reason of sudden increase caused by the increasing of the number of collisions and the number of transmissions, similar to Figure 9. The increase of the number of nodes and distance between nodes increases the number of collisions and the number of transmissions. LMFSim used part of LoRaSim, which set the distance between node randomly based on the Path Loss model [18]. Either using the PoW or LMF platform, the Path Loss model used in the LMFSim is the same.

Figure 13 shows that the greater the number of broadcast domains (N_{broker}), the smaller the total transmission time ($\sum T$) when the number of broadcast domains (N_{broker}) approaches the number of nodes (N_{node}). Increasing the number of broadcast domains (N_{broker}) decreases the number of broadcast domains. This is because LMF does not broadcast on all the nodes, but rather only on the nodes that have the same broadcast domain. When the number of broadcast domains (N_{broker}) gets closer to the number of nodes (N_{node}), the broadcasting process decreases. The transmission time also decreases.

It can be concluded that on the IoT network with the LMF blockchain model, an increase of the number of nodes (N_{node}) will increase the total transmission time ($\sum T$), as shown in Figure 12. The total transmission time ($\sum T$) decreases when the number of broadcast domains (N_{Broker}) approaches the number of nodes (N_{node}), as in Figure 13. Equation (3) is used to measure the average increase of the total transmission time ($T_{average}$) with respect to the number of nodes (N_{node}) or the number of broadcast domains (N_{Broker}).

From Equation (3), the average increase of the total transmission time per node is 0.53%. In addition, the average increase of the total transmission time per additional broadcast domain is 0.27%. It can be concluded that the increase of the number of broadcast domains (N_{Broker}) shortened the time needed for sending compared to increasing the number of nodes (N_{node}).

Furthermore, we calculate the total transmission energy ($\sum E$) obtained from the first and second simulations. A comparison of the total transmission energy and the number of nodes (N_{node}) can be seen in Figure 14. A comparison of the total transmission energy and the number of broadcast domains (N_{Broker}) can be seen in Figure 15.

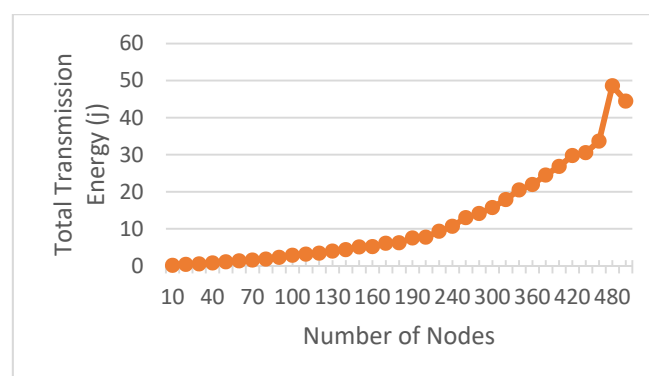


Figure 14. The total transmission energy vs. the number of nodes in the LMF model.

Figure 14 shows that the greater the number of nodes (N_{node}), the higher the total transmission energy ($\sum E$). The energy needed to process the mining and broadcasting of packets to all existing nodes increases with the number of nodes (N_{node}). The increase of the amount of energy is what increases the total transmission energy ($\sum E$). The sudden increase of the total transmission energy ($\sum E$) also happens in Figure 14. This increase happens when the number of nodes (N_{node}) reaches 480. The reason of this sudden increase is the increase of the number of collisions and the number of transmissions. The increase of

the number of collisions and the number of transmissions caused by the random distance between nodes. LMFSim used part of LoRaSim, which set the distance between the nodes randomly based on the Path Loss model [18]. The highest number of collision and transmission happens when the number of miner nodes (N_{miner}) reaches 480.

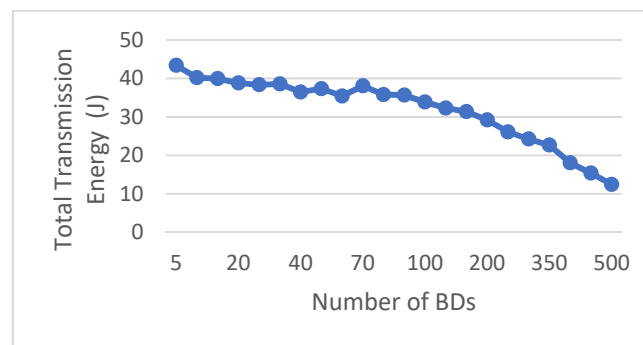


Figure 15. The total transmission energy vs. the number of BDs in the LMF model.

Figure 15 shows that the greater the number of broadcast domains (N_{broker}), the smaller the total energy transmission ($\sum E$) when the number of broadcast domains (N_{broker}) approaches the number of nodes (N_{node}). Increasing the number of broadcast domains (N_{broker}) decreases the number of broadcasts. This is because LMF does not broadcast on all nodes but only on the nodes that have the same broadcast domain. When the number of broadcast domains (N_{broker}) approaches the number of nodes (N_{node}), the duration of the broadcasting process decreases. Therefore, the amount of energy needed also decreases.

It can be concluded that on the IoT network with the LMF blockchain model, an increase of the number of nodes (N_{node}) will increase the total transmission energy ($\sum E$), as shown in Figure 14. The decreasing total transmission energy ($\sum E$) when the number of broadcast domains (N_{Broker}) approaches the number of nodes (N_{node}) is shown in Figure 15. Equation (4) can be used to measure the average increase of the total transmission energy ($E_{average}$) with respect to the number of nodes (N_{node}) or the number of broadcast domains (N_{Broker}).

From Equation (4), the average increase of the total transmission energy per node is 1.73%. Moreover, the average increase of the total transmission energy per additional Broadcast Domain is 0.28%. It can be concluded that the increase of the number of broadcast domains (N_{Broker}) makes the amount of energy needed for transmission smaller than that when the number of nodes (N_{node}) increases.

4.3. Comparison of the PoW and LMF Models on the LoRa IoT Networks

From the simulation, we get the total transmission time ($\sum T$) and the total amount of energy ($\sum E$) for each model. Using these values, we can compare the two blockchain models. The total transmission time ($\sum T$) is obtained from the first and second simulations. A comparison of the total transmission time and the number of nodes (N_{node}) in the PoW and LMF models can be seen in Figure 16. In addition, a comparison of the total transmission time and the number of broadcast domains (N_{Broker}) or the number of miner nodes (N_{Miner}) can be seen in Figure 17.

Figure 16 shows that the greater the number of nodes (N_{node}), the higher the total transmission time ($\sum T$). The increase of the total transmission time ($\sum T$) for the PoW and LMF models is almost the same. When compared to the average increase of the average total transmission time ($T_{average}$) based on Equation (3), the LMF model has a smaller value ($T_{average}$), namely, 0.53%. LMF is more scalable than PoW. When there is an attack on a node, the attack cannot impact a node in another broadcast domain. PoW has only one broadcast domain; therefore, when an attack comes, it will impact all nodes in the system.

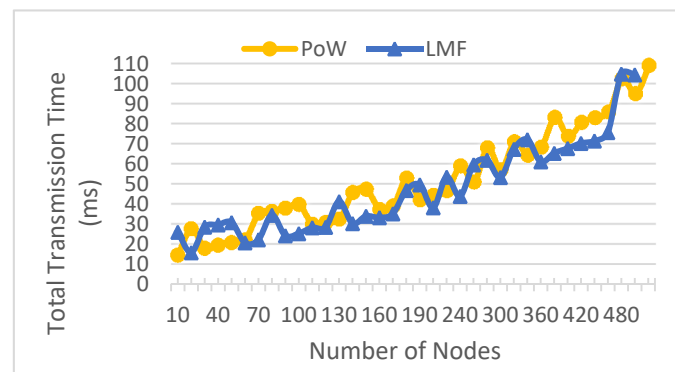


Figure 16. The total transmission time vs. the number of nodes on the PoW and LMF models.

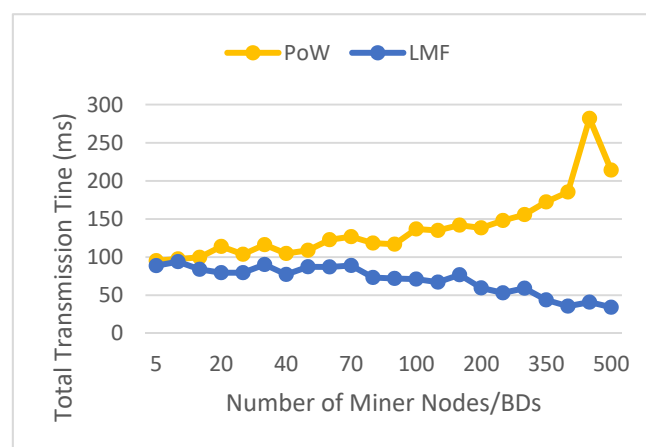


Figure 17. The total transmission time vs. the number of miners/BDs on the PoW and LMF models.

Figure 17 shows a clear difference between the PoW and LMF models. The higher the number of broadcast domains (N_{broker}), the smaller the total transmission time ($\sum T$). The greater the number of miner nodes (N_{miner}), the higher the total transmission time ($\sum T$). When compared to the average increase of the average total transmission time ($T_{average}$) based on Equation (3), the LMF model has a smaller increase of its average total transmission time ($T_{average}$), namely, 0.27%.

It can be concluded that the IoT network with the LMF blockchain model has a smaller impact on the increase of the total transmission time ($\sum T$). This is calculated based on a comparison of the increase of the average total transmission time ($T_{average}$).

Furthermore, the total transmission energy ($\sum E$) is obtained from the first and second simulations. The comparison of the total transmission energy to the number of nodes (N_{node}) in the PoW and LMF models can be seen in Figure 18. The comparison of the total transmission energy to the number of broadcast domains (N_{Broker}) or the number of miner nodes (N_{Miner}) can be seen in Figure 19.

Figure 18 shows that the greater the number of nodes (N_{node}), the higher the total transmission energy ($\sum E$). The increase of the total energy delivered ($\sum E$) between the PoW and LMF models is almost the same. Compared to the increase of the average transmission energy ($E_{average}$) based on Equation (4), the PoW model has a smaller value ($E_{average}$), namely, 1.68%.

Figure 19 shows a clear difference between the PoW and LMF models. The greater the number of broadcast domains (N_{broker}), the smaller the total transmission energy ($\sum E$). When the number of miner nodes (N_{miner}) is greater, the total transmission energy ($\sum E$) will be even greater. When compared to the average increase of the average total transmission energy ($E_{average}$) based on Equation (4), the LMF model has a smaller increase of the average of total transmission energy ($E_{average}$), namely, 0.28%.

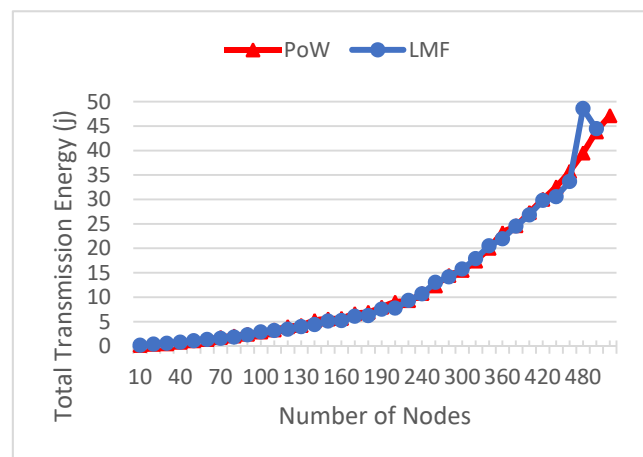


Figure 18. The total transmission energy vs. the number of nodes in the PoW and LMF models.

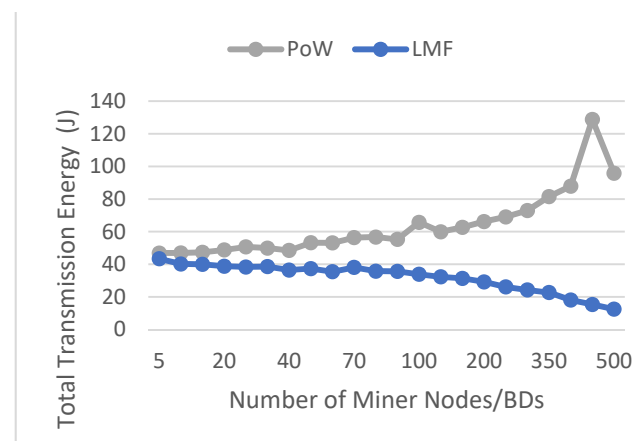


Figure 19. The total transmission energy vs. the total number of miners/BDs on the PoW and LMF models.

It can be concluded that the IoT network with the LMF blockchain model has a smaller impact on the increase of the total transmission energy ($\sum E$) based on the increase of the total average transmission energy ($E_{average}$) if the number of broadcast domains (N_{Broker}) increases. The IoT network with the PoW blockchain model has a smaller impact on the increase of the total transmission energy ($\sum E$) based on the average increase of the total transmission energy ($E_{average}$) if the number of nodes (N_{node}) increases.

Although LMF itself has the advantage of lower latency and energy, it makes the blockchain not as strong as the blockchain on the PoW platform. This is because each broadcast domain node has its own blockchain, which is not owned by another node on a different broadcast domain. To overcome this, two or more broadcast domains can be used to increase the availability if one broadcast domain dies or if a node in the cloud becomes a node or member of each broadcast domain.

4.4. Future Works

As mentioned before, this paper simulated only how the broadcast domain works on the LMF platform and the advantage of separating the broadcast domain from its own fog topology, but it did not simulate the entirety of the LMF platform or consensus. We also have yet to simulate and implement the LMF platform as a prototype in its entirety. We have not tested the encryption mechanism and how to keep all nodes synced in near-real time. The encryption mechanism could be improved upon in the future, for example by combining Software and Hardware certifications as part of Trusted Computing technology [20].

Backup mechanisms for the cloud, proposed in our previous paper [14] and reexplained in detail on Section 2.1, will become the key to keeping nodes and brokers in sync, and ensure that they can take over another broadcast domain when one or some broadcast domains fails/are attacked. The cloud can become another broker who will be a member of every broadcast domain. If one broadcast domain is unavailable, another broker on a different broadcast domain can take over the process with the cloud as its node member, who has backup data from an unavailable node in the failed broadcast domain [13].

The Monitoring and Provisioning Application will be the center to monitor and provision brokers and nodes in Broadcast Domain/Fog Networks and Cloud Nodes, so it still has some security challenges. A performance-oriented Monitoring System can be one of the solutions in the future for Applications to manage, monitor and provide brokers and nodes more securely [21]. By using a Performance Oriented Monitoring System, the Application can monitor the availability, security and capacity of all brokers and nodes safely. The application itself can be strengthened with triple-layered monitoring [22] and Dynamic Security Monitoring as part of a PASSIVE infrastructure [23].

In this paper, we did not discuss the end devices or user devices at the Access Layer in detail. In the future, we will need to consider exploring the possibilities of end-to-end simulation, implementation and security. TPM-based protection can be used to secure end devices with the SecMiLiA library [24] and auto-configurable environment [25].

We also have not yet analyzed the advantage of broadcast domain separation in a country that has a data localization policy. Some countries, such as Indonesia, have a data localization policy which states that data are prohibited from being saved outside the territory of Indonesia [26]. We also have not yet compared this method with another lightweight blockchain platform, such as LSB [13] and FogBus [12]; this will be the focus of one of our future works.

5. Related Works

Many works have discussed the performance evaluation of the blockchain and LoRa network separately. This work may be the first to discuss the performance evaluation of the blockchain on LoRa networks together.

5.1. LoRaWAN Performance Simulation

The parameter used by this work is mainly derived from the LoRaSIM simulator [18]. The simulation run on LoRaSIM was used to choose the best transmission options [18]. There are many works on the LoRaWAN performance evaluation. LoRaWAN performance propagated in an indoor spot [27], Ref. [28] is acquired by measuring area coverage, packet loss, received signal power, the energy consumption of end devices and delays caused by the radio duty cycle. Another study [29] used real measurement to achieve an 80% packet success rate for distances shorter than 5 km and 60% for the distance between 5–10 km. Scalability testing using LoRaSIM was also carried out to prove the scalability of LoRa networks [18] and then reduce the inter-network interference of several LoRa networks [30]. In our previous work, we developed a LoRaSIM simulator to measure the performance of a mobile gateway in the LoRa network for an Intelligent Livestock Monitoring System with a mobile gateway [31]. In this work, we developed an LMFSim based on LoRaSIM to simulate packet transmission and performance evaluation of the Broadcast Domain in the LoRa based Internet of Things. The best parameter from LoRaSIM is used here as a reference, and we used the energy consumption and transmission time as the calculated and compared parameters.

5.2. Multi-Layer Blockchain Based Frameworks for the Internet of Things

Many researchers have been researching blockchain to solve the security problem of the Internet of Things. A decentralization mechanism is a common trait between IoT and blockchain. However, in actual research, the compatibility is not similar in terms of the computing resource and latency. Many researchers tried to provide a platform

or framework that can be used for IoT. Lightweight Scalable Blockchain (LSB) provides lightweight blockchain consensus with light verification to increase latency and minimize computing resources [13]. The performance evaluation of the multi-layer blockchain was presented for LSB [13]. That author proposed the LSB method as a tiered structured blockchain with lightweight verification and distributed management on overlay networks. LSB was simulated using Simpy in [13]. Another work on blockchain performance is the FogBus, which uses fog computing and cloud networks as its method [12]. The difference is that FogBus is simulated using a Raspberry Pi and Dell laptop as its hardware and Java as its software [12], which is close to the real environment. LMF is simulated using LMFSim, a Simpy-based simulator, because it uses hundreds of nodes.

Some researchers proposed a blockchain platform to address privacy, security, fault-tolerance and autonomous behavior [32] and a new authorization framework based on Hyperledger Fabric by enhancing consensus algorithm [33]. Other works to make blockchain implementation possible in IoT [34] turned an IoT network into a multi-layered decentralized network and proposed a multi-level blockchain framework to increase security in IoT [35]. Other work [36] proposed multi-layer security for IoT devices based on distributed blockchain technology by using a Hyperledger Framework for blockchain deployment and verification. Simulations run in [36] showed that a lightweight blockchain framework was more effective than a global traditional blockchain. Combining blockchain with IoT can be highly effective because blockchain provides resilience from attacks and is auditable [37]. All these studies motivated us to build a simulator based on our proposed framework [14] by conducting performance evaluations of energy consumption and transmission time.

The simulator used here is called LMFSim, which is a combination of the LoRaSIM simulator [18] used to simulate LoRa traffic and distribution and the BlockSIM simulator used to simulate the standard PoW blockchain algorithm [19] with an additional LMF algorithm. There are many other simulators in use for simulating blockchain in the IoT, such as Cooja, which is used by LSB [13] and SH-BlockCC [38]. There are also testbed environments using Java, such as like FogBus [12] and BlockEdge [39], and Javascript, such as IBCbAP [40]. LMFSim used Simpy as a basis with Python programming language. Simpy is used because of its popularity in simulating LoRaWAN networks that can emulate nodes in the network and gateways in random points [41].

6. Conclusions

In this research, a simulator has been successfully designed and used to conduct experiments and evaluations of the action mechanism of broadcast domains on the IoT LoRa network with LMF and PoW blockchain models.

The comparisons of the increase of total transmission time ($\sum T$) and total transmission energy ($\sum E$) on the PoW and LMF models show that the results are almost the same when the number of nodes is increased (N_{node}). Regarding the LMF, the average increase of the total transmission time ($\sum T$), which is 0.53%, is smaller and the total transmission energy ($\sum E$), which is 1.73%, is higher than those of PoW. When there is an increase of the number of miner nodes (N_{node}) or the number of broadcast domains (N_{Broker}), the average increase of the total transmission time ($\sum T$), which is 0.27%, is smaller and the total transmission energy ($\sum E$), which is 0.286, is smaller than those of PoW.

Although LMF has the advantage of less latency and energy, this makes the LMF blockchain weaker than the PoW platform. This is because each broadcast domain node has its own blockchain. To overcome this, two or more broadcast domains can be used to increase the availability.

The application of the blockchain on the IoT network can continue to be developed to get the best model or platform from the blockchain, which can increase the security of the IoT network without increasing the latency and required computing resources. LMF itself with its broadcast domain can still be developed further so that it can decrease the latency and computing resources of the IoT network. The LMF model blockchain simulation also needs to be further developed using a modeling consensus, not just the broadcast domain

mechanism, so that the results will be closer to reality before the model can be applied to the real world.

Author Contributions: Conceptualization, M.Y.A.S.; methodology, M.Y.A.S. and R.F.S.; software, M.Y.A.S.; validation, M.Y.A.S. and R.F.S.; formal analysis, M.Y.A.S. and R.F.S.; investigation, M.Y.A.S.; resources, M.Y.A.S.; data curation, M.Y.A.S.; writing—original draft preparation, M.Y.A.S.; writing—review and editing, M.Y.A.S. and R.F.S.; visualization, M.Y.A.S. and R.F.S.; supervision, R.F.S.; project administration, R.F.S.; funding acquisition, R.F.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by University of Indonesia under the Q1Q2 Grant Number NKB-0321/UN2.R3.1/HKP.05.00/2019.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available from the corresponding author on reasonable request.

Acknowledgments: This work is supported by the University of Indonesia under the Q1Q2 Grant Number NKB-0321/UN2.R3.1/HKP.05.00/2019.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gubbi, J.; Buyya, R.; Marusic, S.; Palaniswarni, M. Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Gener. Comput. Syst.* **2013**, *29*, 645–1660. [\[CrossRef\]](#)
2. Atzori, L.; Iera, A.; Morabito, G. The Internet of Things: A survey. *Comput. Netw.* **2010**, *54*, 2787–2805. [\[CrossRef\]](#)
3. Suryadevara, N.; Mukhopadhyay, S.C. Internet of things: Challenges and opportunities. *Smart Sens. Meas. Instrum.* **2014**, *9*, 1–17.
4. Pacelle, M. Radar O'Reilly. 2014. Available online: <http://radar.oreilly.com/2014/04/3-topologies-driving-iot-networking-standards.html> (accessed on 6 October 2019).
5. Bonderud, D. Security Intelligence. 2016. Available online: <https://securityintelligence.com/news/leaked-mirai-malware-boosts-iot-insecurity-threat-level> (accessed on 6 October 2019).
6. Tawalbeh, L.; Muheidat, F.; Tawalbeh, M.; Quwaider, M. IoT Privacy and Security: Challenges and Solution. *Appl. Sci.* **2020**, *10*, 4102. [\[CrossRef\]](#)
7. Ali, M.S.; Dolui, K.; Antonelli, F. IoT data privacy via blockchains and IPFS. In Proceedings of the Seventh International Conference on the Internet of Things (IoT '17), New York, NY, USA, 22–25 October 2017.
8. Kouicem, D.E.; Bouabdallah, A.; Lakhlef, H. Internet of things security: A top-down survey. *Comput. Netw.* **2018**, *141*, 199–221. [\[CrossRef\]](#)
9. Alphan, O.; Amoretti, M.; Dall'Asta, C.T.S.; Ferrari, D.A.G.; Rousseau, F.; Tourancheau, B.; Veltri, L.; Zanichelli, F. IoTChain: A blockchain security architecture for the Internet of Things. In Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), Barcelona, Spain, 15–18 April 2018.
10. Zhou, L.; Wang, L.; Sun, Y.; Lv, P. BeeKeeper: A Blockchain-Based IoT System With Secure Storage and Homomorphic Computation. *IEEE Access* **2018**, *6*, 43472–43488. [\[CrossRef\]](#)
11. Dorri, A.; Kanhere, S.S.; Jurdak, R. Towards an Optimized Blockchain for IoT. In Proceedings of the Second International Conference of Internet-of-Things Design and Implementation, Pittsburgh, PA, USA, 18–21 April 2017.
12. Tuli, S.; Mahmud, M.; Tuli, S.; Buyya, R. A Blockchain-based Lightweight Framework for Edge and Fog Computing. *J. Syst. Softw.* **2019**, *154*, 22–36. [\[CrossRef\]](#)
13. Dorri, A.; Kanhere, S.S.; Jurdak, R.; Gauravaram, P. LSB: A Lightweight Scalable Blockchain for IoT Security and anonymity. *J. Parallel Distrib. Comput.* **2019**, *134*, 180–197. [\[CrossRef\]](#)
14. Saputro, M.Y.A.; Sari, R.F. Securing IoT Network using Lightweight MultiFog (LMF) Blockchain Model. In Proceedings of the International Conference on Electrical Engineering, Computer Science and Informatics (EECSI 2019), Bandung, Indonesia, 18–20 September 2019.
15. Mekki, K.; Bajic, E.; Chaxel, F.; Meyer, F. A comparative study of LPWAN technologies for large-scale IoT deployment. *ICT Express* **2019**, *5*, 1–7. [\[CrossRef\]](#)
16. Foubert, B.; Mitton, N. Long-Range Wireless Radio Technologies: A Survey. *Future Internet* **2020**, *12*, 13. [\[CrossRef\]](#)
17. Castells, P.; Cruz, G.; Masaki, T.; Castelan, C.R. World Bank Blogs. World Bank. 2020. Available online: <https://blogs.worldbank.org/developmenttalk/expanding-mobile-broadband-coverage-lifting-millions-out-poverty> (accessed on 2 April 2021).
18. Bor, M.C.; Roedig, U.; Voigt, T.; Alonso, J.M. Do LoRa Low-Power Wide-Area Networks Scale? In Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, Floriana, Malta, 13–17 November 2016.

19. Pandey, S.; Ojha, G.; Shrestha, B.; Kumar, R. BlockSIM: A practical simulation tool for optimal network design, stability and planning. In Proceedings of the 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), Seoul, Korea, 14–17 May 2019.
20. Munoz, A.; Mafia, A. Software and hardware certification techniques in a combined certification model. In Proceedings of the 11th International Conference on Security and Cryptography (SECRYPT), Vienna, Austria, 28–30 August 2014.
21. Muñoz, A.; Gonzalez, J.; Maña, A. A Performance-Oriented Monitoring System for Security Properties in Cloud Computing Applications. *Comput. J.* **2012**, *55*, 979–994. [[CrossRef](#)]
22. Toutouh, J.; Muñoz, A.; Neschachnow, S. Evolution Oriented Monitoring oriented to Security Properties for Cloud Applications. In Proceedings of the 13th International Conference on Availability, Reliability and Security (ARES 2018), Hamburg, Germany, 27–30 August 2018.
23. Muñoz, A.; Harjani, R.; Maña, A.; Díaz, R. Dynamic Security Monitoring and Accounting for Virtualized Environments. In *Secure and Trust Computing, Data Management, and Applications STA 2011, Communications in Computer and Information Science*; Lee, C., Seigneur, J.M., Park, J.J., Wagner, R.R., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; Volume 187.
24. Muñoz, A.; Maña, A. TPM-based protection for mobile agents. *Secur. Commun. Netw.* **2010**, *4*, 45–60. [[CrossRef](#)]
25. Lopez, J.; Mana, A.; Munoz, A. A Secure and Auto-configurable Environment for Mobile Agents in Ubiquitous Computing Scenarios. In Proceedings of the Third International Conference on Ubiquitous Intelligence and Computing, Wuhan, China, 3–6 September 2006; Volume 4159, pp. 977–987.
26. Deradjat, A.; Timur, M. ABNR Law. 2019. Available online: https://www.abnrlaw.com/news_detail.php?send_news_id=366&year=2019 (accessed on 8 September 2020).
27. Neumann, P.; Montavond, J.; Noel, T. Indoor Deployment of Low-Power Wide Area Networks (LPWAN). In Proceedings of the IEEE 12th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), New York, NY, USA, 17–19 October 2016.
28. Petäjajarvi, J.; Mikhaylov, K.; Hämmäläinen, M.; Iinatti, J. Valuation of LoRa LPWAN Technology for Remote Health and Wellbeing Monitoring. In Proceedings of the 10th International Symposium on Medical Information and Communication Technology (ISMICT), Worcester, MA, USA, 20–23 March 2016.
29. Petajarvi, J.; Mikhaylov, K.; Roivainen, A.; Hanninen, T.; Pettissalo, M. On the Coverage of LPWANs: Range Evaluation and Channel Attenuation Model for LoRa Technology. In Proceedings of the 14th International Conference on ITS Telecommunications (ITST), Copenhagen, Denmark, 2–4 December 2015.
30. Bor, M.C.; Roedig, U.; Alonso, J. Mitigating Inter-Network Interference in LoRa Networks. In Proceedings of the 2017 International Conference on Embedded Wireless Systems and Networks, Uppsala, Sweden, 20–22 February 2017.
31. Ikhsan, M.G.; Saputro, M.Y.A.; Arji, D.A.; Harwahyu, R.; Sari, R.F. Mobile LoRa Gateway for Smart Livestock Monitoring System. In Proceedings of the IEEE International Conference on Internet of Things and Intelligence System (IOTAIS), Bali, Indonesia, 1–3 November 2018.
32. Pahl, C.; El Ioini, N.; Helmer, S. A Decision Framework for Blockchain Platforms for IoT and Edge Computing. In Proceedings of the International Conference on Internet of Things, Big Data and Security (IoTBDS), Funchal, Portugal, 19–21 March 2018.
33. Klaokliang, N.; Teawtim, P.; Aimtongkham, P.; So-In, C.; Niruntasukrat, A. A Novel IoT Authorization Architecture on Hyper-ledger Fabric with Optimal Consensus Using Genetic Algorithm. In Proceedings of the Proceedings of the 2018 Seventh ICT International Student Project Conference (ICT-ISPC), Nakhon Pathom, Thailand, 11–13 July 2018.
34. Chendeb, N.; Khaled, N.; Agoulmine, N. Integrating Blockchain with IoT for a Secure Healthcare Digital System. In Proceedings of the 8th International Workshop on ADVANCES in ICT Infrastructures and Services (ADVANCE 2020), Cancun, Mexico, 27–29 January 2020.
35. Mbarek, B.; Jabeur, N.; Pitner, T.M. Multilevel blockchain system for IoT, Ubiquitous Comput. *Pers. Ubiquitous Comput.* **2019**, *11*, 1–8.
36. Pajooh, H.H.; Rashid, M.; Alam, F.; Demidenko, S. Multi-Layer Blockchain-Based Security Architecture for Internet of Things. *Sensors* **2021**, *21*, 772. [[CrossRef](#)] [[PubMed](#)]
37. Jesus, E.F.; Chicarino, V.R.L.; de Albuquerque, C.V.N.; Rocha, A.A.D. A Survey of How to Use Blockchain to Secure Internet of Things and the Stalker Attack. *Secur. Commun. Netw.* **2018**, *2018*, 27. [[CrossRef](#)]
38. Singh, S.; Ra, I.H.; Meng, W.; Kaur, M.; Cho, G.H. SH-BlockCC: A secure and efficient Internet of things smart home architecture based on cloud computing and blockchain technology. *Int. J. Distrib. Sens. Netw.* **2019**, *15*, 1–18. [[CrossRef](#)]
39. Yavari, M.; Safkhani, M.; Kumari, S.; Kumar, S.; Chen, C.-M. An Improved Blockchain-Based Authentication Protocol for IoT Network Management. *Secur. Commun. Netw.* **2020**, *2020*, 16. [[CrossRef](#)]
40. Kumar, T.; Harjula, E.; Ejaz, M.; Manzoor, A.; Porombage, P.; Ahmad, I.; Liyanage, M.; Braeken, A.; Ylianttila, M. BlockEdge: Blockchain-Edge Framework for Industrial IoT Networks. *IEEE Access* **2020**, *8*, 54166–54185. [[CrossRef](#)]
41. Marini, R.; Mikhaylov, K.; Pasolini, G.; Buratti, C. LoRaWANSim: A Flexible Simulator for LoRaWAN Networks. *Sensors* **2021**, *21*, 695. [[CrossRef](#)] [[PubMed](#)]