Securing IoT Network using Lightweight Multi-Fog (LMF) Blockchain Model

Muhammad Yanuar Ary Saputro
Department of Electrical Engineering
University of Indonesia
Kampus Baru UI Depok
m.yanuar71@ui.ac.id

Riri Fitri Sari
Department of Electrical Engineering
University of Indonesia
Kampus Baru UI Depok
riri@ui.ac.id

***Abstract*— Security is one of the most important issues in the Internet of Things (IoT). The Mirai botnet case in September 2016 revealed a serious vulnerability in IoT devices. Researchers try to mitigate the issues using several approaches. One of them uses Blockchain for the solution. At first, the integration of the Blockchain on IoT seems promising. However, there are problems in resource consumption and latency. Several solutions emerge to make Blockchain uses low resource consumption i.e., LSB and FogBus. Unfortunately, each solution has its weaknesses. FogBus has a weakness in integrity, whereas LSB has a weakness in its availability when an attack occurs on a broker. We introduce Lightweight Multi-Fog (LMF) Blockchain Model to increase availability in the LSB model. The main idea is increasing the integrity availability by splitting location based on Broadcast Domains while using Fog Computing on each Broadcast Domain. An attack in some Broadcast Domain cannot impact transactions and process in other Broadcast Domain and each Broadcast Domain have its separate transaction and process. LMF enhances the integrity and availability of the Light Blockchain Model. However, it still requires simulations in the future to get a better understanding of LMF performance, resource consumption, and latency.**

Keywords*— *Blockchain, Fog Computing, IoT, Lightweight, Network.

I. INTRODUCTION

Internet of Things (IoT) is a recent widely used technology that makes everything possible to connect to the internet and communicate with each other. IoT sometimes refers to Machine to Machine (M2M) [1]. It is slightly different because there is not only communication among machines but also people involved in communication. Another difference is IoT uses sensor technology and wireless communication with low power usage [2]. The most interesting part of this technology is not limited to the automation industry, but also about the way we live in our home. The technology is called a smart home. One example of the implementation is to make the garage door automatically opened when people come home [3].

There are security risks in rapid IoT implementation [4]. One of the biggest issues is the Mirai botnet attack in September 2016. Mirai attack in 2016 is the biggest problem in IoT. This reveals a serious vulnerability in IoT devices. Mirai uses BASHLITE in a DDoS attack on Krebs on Security website on September 20, 2016. Ars Technica also reports several attacks on the French website [5].

This attack also happened to one DNS Service Provider, Dyn, on October 21, 2016. They are attacked by Mirai malware that is installed on a large number of IoT devices. Therefore, big websites i.e. GitHub, Twitter, Reddit, Netflix, Airbnb, and many others are inaccessible. At the end of November 2016, around 900,000 Arcadya's routers at Deutsche Telekom are also inaccessible during hacking attempts using failed TR-064. This is a variant of Mirai malware that causes Internet connectivity problems [6]. Mirai attack started by the attacker by taking control of Control Server and then the attacker installs BASHLITE program on the server to launch massive DDoS attack on the network like in Figure 1. The main reason IoT devices easier to attack, due to the lack of security and the patches are rarely released.

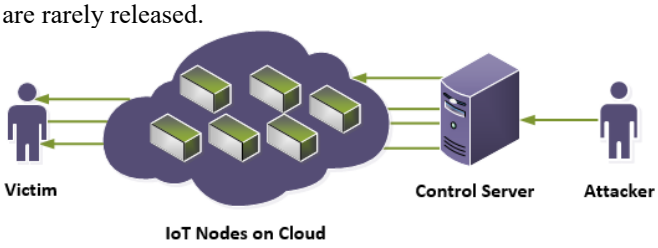

Fig. 1. Mirai Botnet Attack Flows [6]

According to many surveys conducted by the researcher, they find that not only cameras are vulnerable [7]. CCTV systems and cable boxes are also vulnerable to cyber-attacks. The vulnerability of the IoT device is also related to the lack of awareness of IoT devices manufacturers in designing their devices [7].

There are many solutions provided by researchers to mitigate this issue. One of them is to integrate the Blockchain with IoT technology [8], [9], [10]. Blockchain is introduced by Satoshi Nakamoto in 2008. It is introduced as the technology behind BitCoin as a digital currency system. It uses peer to peer communication similar to BitTorrent. Blockchain consists block of data that is connected like a chain. Everyone can be a miner, an entity that has the authority to solve the cryptography puzzle, known as *Proof of Work* (PoW) and add a new block to the Blockchain. When a transaction occurs, transaction information will be broadcast to the entire network. Then each miner validates and signs the transaction and adds the transaction data to their block [8]. Many solutions have been provided by researchers. However, Blockchain is predicted as focused research to secure IoT [11]

The new Blockchain-based Multi-Layer Secure Network Model is proposed as a combination of the centralized and decentralized IoT Network Model and enhances its security using the Blockchain [8]. This mechanism requires large computational resources that are very limited on IoT devices [12]. Most IoT devices have low power [13]. Another thing, the time that is needed to complete a transaction by Blockchain is rather huge, for example in Bitcoin can take up to 30 minutes for transaction to be confirmed. This delay is not acceptable for IoT communication [14]. Another research also notes the

same problem when integrating the Blockchain into IoT [15], [16].

Reducing the computing resources and reducing delays for completing transactions are the only way that the Blockchain can be used in the IoT network. Several researchers [17] try to use the Blockchain on IoT. However, instead of using it on IoT transactions, they use it only as an authentication and authorization scheme on IoT. This solution does not solve the problems that occur when integrating the Blockchain into the IoT. It is only applying the Blockchain as a AAA mechanism in IoT.

Another researcher [18] propose an Ethereum-based algorithm to integrate the Blockchain in IoT called BeeKeeper. However, the latency is still categorized as high with more than 10 seconds for block intervals. Other research proposes a Fog and Cloud-based algorithm [10] called FogBus. It integrates the Blockchain into IoT, but it is lack of detailed security.

The most notable research to integrate the Blockchain into IoT is to implement a new consensus algorithm called Lightweight Scalable Blockchain (LSB). LSB uses direct and indirect evidence to reduce the latency used by transactions that must be completed. Time-based consensus algorithms are placed rather than Proof of Work (PoW) or Proof of Stake (PoS) [14]. Miners or Brokers in an overlay network are called Overlay Broker Managers (OBM) and Miners or Brokers in Local Networks are called Local Broker Managers (LBM). OBM uses asymmetric encryption, whereas LBM uses symmetric encryption. This last consensus still has availability issues. One of the issues is performance degradation to all nodes when multiple nodes are attacked.

LSB and FogBus are the most complete solutions to solve Blockchain and IoT integration problems using different approaches. However, both have their weaknesses. LSB topology does not elaborate on the details of communication among OBMs at the network level. Since all OBMs broadcast each other in overlay networks, it can be assumed that all OBMs are in the same broadcast domain. Therefore, when an OBM is attacked, it is possible that the attack also targets other OBMs in the same overlay network. Whenever there is a DDoS attack occurs to an OBM, not only attack a node, the transaction verification services can be limited to several unisolated OBMs and nodes. An additional intermediate layer can be used to reduce this issue by dividing the broadcast domain. Therefore, attacks on one broadcast domain will not affect other broadcast domains. However, the FogBus framework is not tested against several attack scenarios. The latency is still 50-300% higher than without using the Blockchain [10].

We introduce Lightweight Multi Fog (LMF) Model to mitigate the IoT security risk. We integrate the LSB and FogBus algorithm. To mitigate both possible weaknesses, we implement the LSB mechanism in the FogBus framework and break the FCN into separate broadcast domains. It reduces the delay in FogBus and increases the security factor by separating the broadcast domain in LSB without losing the advantages of the Blockchain mechanism.

II. LIGHTWEIGHT MULTI-FOG BLOCKCHAIN

In this section, we discuss the Lightweight Multi-Fog Design in detail. We begin by defining three fundamental concepts:

- *Transaction*: Typical protocol used for information communication flow before transmitting data in the network among each node.
- *Broadcast Domain*: Network Domain where only all nodes in the same Domain receive all packet broadcast from each node.
- *Broker*: A node in the same broadcast domain that acts as manager. This node is responsible for managing blockchain transactions stored on each node, also verify and authorize the transaction.

A. Lightweight Multi Fog (LMF) Blockchain Architecture

FogBus [10] categorizes each Layer based on three technologies i.e., IoT Layer, Fog Layer, and Cloud Layer. While LSB [14] does not categorize each layer. Therefore, it is difficult to determine the functions that run on each layer. Lightweight Multi Fog (LMF) uses a different layering system. LMF uses a function to distinguish each layer. LMF consists of four layers i.e., Access Layer, Network Layer, Computing Layer, and Application Layer, presented in Figure 2.

Access Layer is the lowest layer that consists of IoT devices and sensors. This layer has a connection to the Internet or private networks. Hence, LMF can be implemented on both public or private networks. In public networks, each device and sensor must be connected to the internet. It usually applies to Smart Public Applications. In private networks, each device or sensor must be able to reach the gateway in the Network Layer. Since the data are not exposed to the public network, it usually applies to Intelligent Industrial Systems.

Network Layer is the layer that runs the network function on the LMF architecture. The Network Layer functions as a gateway. It routes data to Blockchain Broker and Blockchain Nodes. The Network Layer also acts as a gateway for each Broadcast Domain. Since each Broadcast Domain has one Broker and several Nodes, the Network Layer has at least one router. Each Broadcast Domain can also be represented as City, Province or Country. Hence, the Broadcast Domains is equal to the number of Cities, Provinces or Countries where the service is implemented.

Compute Layer consists of at least one Blockchain Broker and several Blockchain Nodes in every Broadcast Domain. By default, the nodes only process transactions from its Broker in the same Broadcast Domain. During Broker unavailability, any resource-available node in the same broadcast domain will take over as a new Broker. Whenever there are no candidate nodes available in one broadcast domain, a node or broker in another Broadcast Domain will take over as a new Broker.

Application Layer is the upper layer of LMF. It consists of several servers that host applications and storage. Transaction data is stored and processed by the application in this layer.

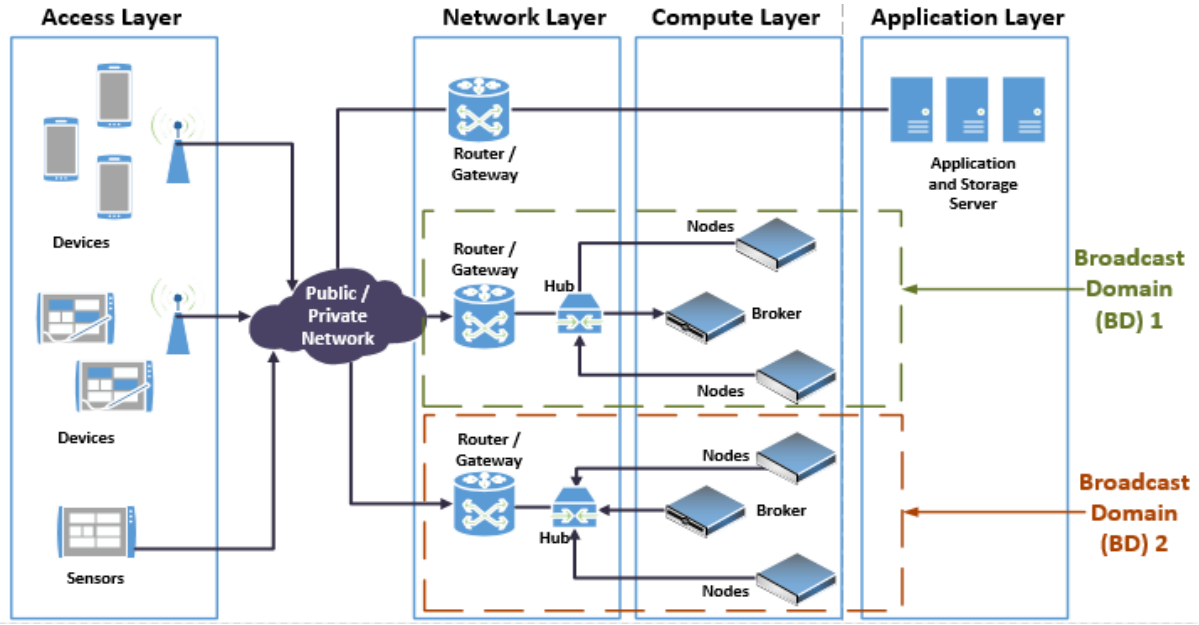

Fig. 2. Lightweight Multi Fog (LMF) Architecture

B. Communication Flow

The LMF Network Model uses Broadcast Domains to separate each zone. Zones represent cities or countries. This implementation model is used to reduce massive DDoS attacks on Broker Nodes. Once a DDoS attack occurs in one or more Brokers in their Broadcast Domains, the attack will not affect Brokers in other Broadcast Domains.

1) Broadcast Domain Selection Process

Each Broadcast Domain has its Nodes that act as Brokers. Other nodes act as compute layers. All of these nodes do not communicate with nodes in other Broadcast Domain nodes. Except when a Broker fails but there are no capable nodes to become a Broker in their Broadcast Domain.

Algorithm 1 describes the procedure for the broker selection process. Recall all nodes are stayed on the Blockchain, each of them will have a 'Broker Readiness Status'. The node that has the criteria to become a broker will have a value of 'TRUE'. At first, the condition of the node status (*X.Condition*) will be determined whether up or down (line 2). Then the value of the Broadcast Domain (*X.BD*) will be determined (line 3). If several nodes have the same Broadcast Domain value, the node's capacity (*X.Capacity*) will be determined next (line 4). Only a sufficient capacity node can become a Broker (line 5,11). When there are no available and capable nodes on the same Broadcast Domain, a node in another Broadcast Domain will be selected as Broker (line 9). A node with the highest capacity will be selected as a Broker for related Broadcast Domains.

Algorithm 1 Broker Readiness Status.

Input: Nodes (X), the Total Number of Nodes (j)
Output: True or False

```
1.    for (i<=j)
2.     if (X(i).Condition=1) then
3.     if (X(i+1).BD=X(i).BD) then
4.     if (X(i).Capacity==1)
5.     return True;
6.     else
7.     return False;
8.     end if
9.     else
10.    if (X(i).Capacity==1)
11.    return True;
12.    else
13.    return False;
14.    end if
15.    end if
16.    else
17.    return False;
18.    end if
19.   end
```

2) Process and Traffic Flow

The mechanism of LSB is the applied process and traffic flow. Each node in the Broadcast Domain has its own Public Key (PK). Each node will generate a unique PK for each transaction. Each block consists of the requester's hash PK and the target's hash PK for this transaction. It also consists of the requester's hash PK for the next transaction [14]. This mechanism ensures that the next transaction is valid. It is done by comparing the requester's PK on the next transaction with the requester's PK that already stored in the previous transaction.

A Broker also communicates with each Broker on different Broadcast Domains. This communication validates the transaction using direct and indirect evidence mechanisms [14]. This mechanism will reduce the time in the verification process. However, unlike LSB, LMF only

stores the blocks in the local Broadcast Domain. Transactions that come to a Broadcast Domain, will not be stored in a node on a different Broadcast. Every Broadcast Domain is possible to have a different Blockchain.

Since each Broadcast Domain has its Blockchain, LMF has a backup mechanism using Cloud resources. Hence, when all nodes on Broadcast Domains unavailable, the blocks are still stored on Cloud Storage. There is a difference between the Cloud and Non-cloud Node. Each node can only have one Broker on their Broadcast Domain. However, the Cloud becomes a node for each Broker on all Broadcast Domains because it acts as a backup node.

The traffic flow in LMF can be explained in Figure 3. The communication between a Broker and Nodes uses Layer 2 Network Protocol. Whereas the communication among Brokers in each Broadcast Domain is using Layer 3 Network Protocol i.e., OSPF, BGP, etc.

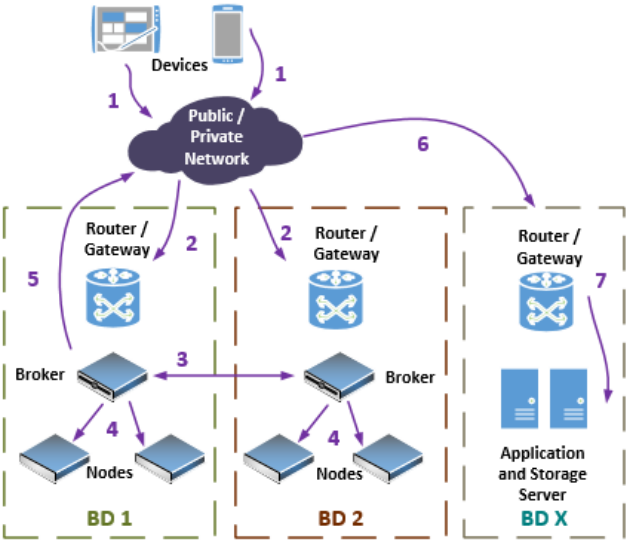

Figure 3. Lightweight Multi Fog (LMF) Data Flow

Figure 3 describes the Communication Flow of Devices or sensors to the Nodes and Servers as follows:

1. Traffic from Devices sent to public/private networks uses the Routing Protocol;
2. Traffic from devices is designated to the nearest gateway in their location, according to their zone, city or country;
3. Traffic is checked and verified by a Broker in the nearest Broadcast Domain. Then it will be compared to another Broker located in other Broadcast Domains, to verify whether valid or not;
4. Data is calculated and stored by the number of available nodes in the local Broadcast Domain;
5. Data is also backed up in a Cloud Server. A Broker communicates to Storage servers using asymmetric encryption and verification through public/private networks;
6. Communication is transferred to the Gateway where Storage and Application Servers are located. The Storage Server and Application Servers are located in a different Broadcast Domain;
7. Transactions and blocks are backed up on the Cloud Storage Server.

C. Transaction Flow Mechanism

The two main transaction flows in LMF are Store Flow and Access Flow.

1) Store Transaction

Using the LSB mechanism [14], transaction flows on the LMF are almost the same, except for storage locations and backup mechanisms. In LSB [14], users can store data locally or in the cloud. They also have local brokers and nodes in each LBM. LMF is designed so that it can be used generally in IoT scenarios. Hence, it does not have a local broker like LSB, it uses Fog computing instead. Every Transaction and Block in the LMF is stored on the Broadcast Domain Nodes located near the Device. Another thing, LMF has a backup mechanism, which is a transaction and block that is also stored on Cloud Storage if all the nodes and broker in a Broadcast Domain are being attacked.

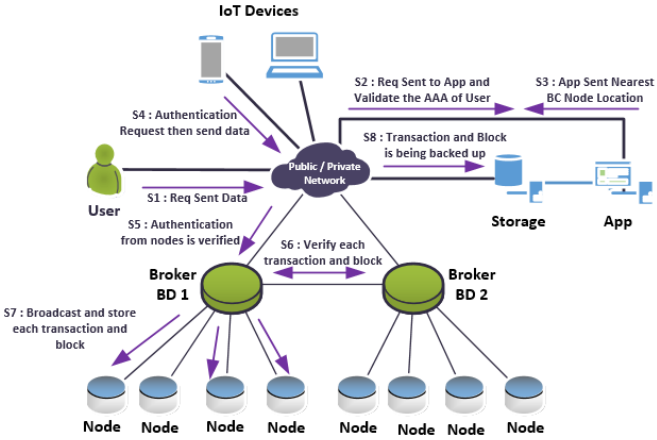

Figure 4. Store Transaction Flow

Figure 4 describes the flow of store transactions in the LMF. When a user or device wants to store the transaction data, the user will be authenticated and authorized (S1, S2). Then the Application checks the nearest available Broker using the Public IP or Private IP Database stored previously in the Application (S3). Then the transaction data will be authenticated by a Broker using asymmetric encryption by validating their Public Keys. It is being validated by all Network Brokers (S5, S6). After the transaction is authorized and validated using the LSB mechanism, the transaction and block are stored in the node on the same Broadcast Domain (S7). The data are also backed up to the Cloud Storage Server (S8).

2) Access Transaction

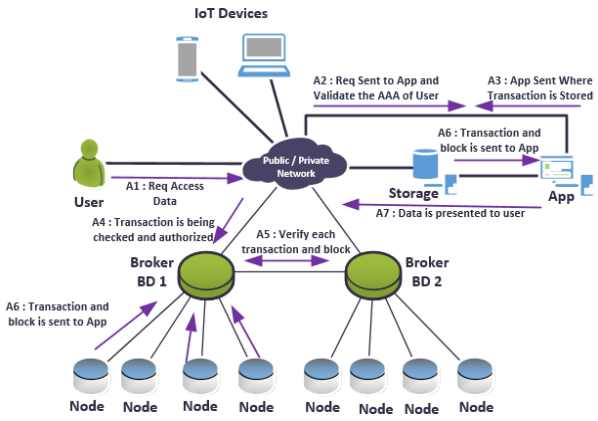

Figure 5. Access Transaction Flow

Figure 5 describes the flow of access transactions in the LMF. When a user wants to access data, that user will be authenticated and authorized (A1, A2). Then the Application forwards the request to the relevant Broker where the transaction data is stored (A3). A request to access a transaction is authenticated by Brokers using asymmetric encryption by validating the application's Public Key and are being validated by all Brokers in the Network (A4, A5). After the request to access the transaction is authorized and validated using the LSB mechanism, the transaction and block from nodes are sent to the Application Server (A6). The Application Server presents data to the User (A7).

III. ANALYSIS AND DISCUSSION

This section provides a security analysis of LMF design, based on the Security Triad i.e., Confidentiality, Integrity, and Availability [19].

1) Confidentiality

LMF is designed based on the combination of LSB [14]for security and lightness, and FogBus [10] for Fog computing and scalability. The communication among components i.e., Devices and a Broker, a Broker and Nodes, a Broker and Cloud Storage, a Broker, and Application, is encrypted using asymmetric encryption. Each Node, Broker, Device, Storage, and Application has its own Public Key (PK). The broker will validate the requester's PK with the hash of PK of the previous block. It will be verified by another broker using direct or indirect evidence [14]. Using this mechanism, each transaction is quite confidential for each request. Only verified and authorized PK can store or access the transactions.

User privacy is protected using changeable PK that is uniquely generated for each transaction. The stored transactions are encrypted using the requester's PK. This mechanism ensures anonymity and privacy. Hence, no one knows the requester's real identity for each transaction.

2) Integrity

Protection against data tampering and fake transactions applied in LMF using hashes in other fields. LMF consists of two headers, transaction headers and block headers. The block header consists of the previous hash transaction and verification signature. The transaction header consists of the next transaction hash. Before a transaction can be validated, the previous transaction hash must be the same as the hash included in the block header. Then the hash in the transaction header stored in the Blockchain to be used in the verification of the next transaction [14]

Broadcast Domain Separation. There is a difference between LMF and LSB in terms of the use of period-consensus mechanisms. The consensus period is used when fake transactions try to access data in the Blockchain. A Broker is limited to one block that can be generated during a period of consensus-period intervals [14]. In LMF, by using Broadcast Domain Separation, only Broker on the same Broadcast Domain that can store and access the transaction and requestor transaction only stored on the Broadcast Domain nearest their location. In case of attacks, the attacks on some Broadcast Domain cannot impact the transactions and processes in other Broadcast Domain, because they cannot store the transaction in nodes on different Broadcast Domain.

Location Verification. Since the LMF does not use a consensus period, the LMF mitigates this issue by verifying the location of the request. A Requester cannot store transactions in another Broadcast Domain when there are available Broker and Nodes in the nearest Broadcast Domain. An attempt to make a transaction to another Broadcast Domain instead of the nearest Broadcast Domain can be categorized as fake transactions. Except when there are no Brokers available in their nearest Broadcast Domain.

Transaction Separation. LMF also has protection on Broadcast Domains where the Cloud Storage Server and Application server are located. Only brokers can request to store data on the Cloud Storage Server. There is no direct access from the Users to the Cloud Storage Server. Broadcast Domain separation mechanism in LMF can also comply with "data localization regulation" that was implemented in some countries since each Broadcast Domain did not store transaction data to another Broadcast Domain.

3) Availability

LMF uses the advantages of FogBus and LSB for its availability. On LSB, when a Broker is not available, the Nodes will choose another Broker to be associated with [14]. Whereas in FogBus, when a Broker is not available, each worker node can become a Broker [10]. LMF combines both availability mechanisms.

LMF Fault Tolerance. Once the existing Broker is not available, the node in the same Broadcast Domain will become Broker if available and have sufficient resources. If there are no available nodes in the same Broadcast Domain, Broker or nodes from nearest Broadcast Domain will take over the nodes with no Broker in their Broadcast Domain. This fault-tolerance mechanism ensures availability in the LMF Blockchain Model while separating Broadcast Domains to represent each City, Province or Country.

Broadcast Domain Separation. Broadcast Domain Implementation is the main advantage offered by LMF in terms of DDoS Attack on Broker or OBM in terms of LSB. On LSB, all Brokers or OBMs are part of the same Overlay Network, which is in the same Broadcast Domain. Although LSB has a period-consensus mechanism, this mechanism will limit the transaction process when an attack occurs. The number of brokers available in LMF is smaller than the number of brokers available in LSB. It is because of the separation of broadcast domains. Hence, this mechanism will increase latency and time for transactions to be processed.

LMF Backup Mechanism. LMF also has a backup mechanism, by using Cloud-nodes that consist of transactions stored on the backup storage. The transaction that is previously stored on the failed nodes or problem Broadcast Domain due to outage or attacks can be accessed and verified by another Broker and nodes in different Broadcast Domain.

IV. CONCLUSION

The proposed Lightweight Multi-Fog (LMF) Blockchain, that integrates the LSB and FogBus algorithms is proposed to reduce delays that appear in FogBus and improve availability and integrity by separating the Broadcast Domain. LMF increases integrity by using the Broadcast Domain separation model. Broadcast Domain is separating transaction and process on each Broadcast Domain. LMF also has a location verification mechanism to make sure the requestor transactions are processed in their nearest Broadcast Domain and protect Brokers from unauthorized transactions using the location checking mechanism.

LMF is expected can increase availability by processing the transaction on the nearest Broadcast Domain, decrease the delay, have fault-tolerance mechanism combined from LSB and FogBus and Backup mechanism in Cloud. So, when an attack on a Broadcast Domain happens, it cannot impact transaction and process in another Broadcast Domain. Each Broadcast Domain have a different chain.

LMF is not yet implemented and tested for performance, resource consumption, and latency. So, future work is needed to simulate the LMF model and compare it with FogBus and LSB. Simulation or Implementation can be done in the future to get a better understanding of performance, resource consumption and latency of LMF. SDN and NFV technology can be implemented in LMF Blockchain IoT to create more efficient nodes and processes [20].

ACKNOWLEDGMENT

The authors would like to thank the University of Indonesia for financial support for this research under the Pit9 Grant, under the contract number: NKB-0072/UN2.R3.1/HKP.05.00/2019.

REFERENCES

[1] S. C. M. a. N. Suryadevara, "Internet of things: Challenges and opportunities," *Internet of Things,* no. Springer, pp. 1-17, 2014.

[2] R. B. S. M. S. M. P. J. Gubbi, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Generation Computer Systems,* vol. 29, pp. 1645 - 1660, 2013.

[3] A. I. G. M. Luigi Atzoria, "The Internet of Things: A survey," *Computer Networks,* vol. 54, pp. 2787 - 2805, 2010.

[4] K. S. M. A. Khan, "IoT security: Review, blockchain solutions, and open challenges," *Future Generation Computer Systems,* vol. 82, pp. 395-411, 2018.

[5] D. Bonderud, "securityintelligence.com," 4 October 2016. [Online]. Available: https://securityintelligence.com/news/leaked-mirai-malware-boosts-iot-insecurity-threat-level/. [Accessed 10 September 2018].

[6] B. Krebs, "krebsonsecurity.com," 30 November 2016. [Online]. Available: https://krebsonsecurity.com/2016/11/new-mirai-worm-knocks-900k-germans-offline/. [Accessed 10 September 2018].

[7] C. B.-D. Elizabeth LaGreca, "Survey on the Insecurity of the Internet of Things," in *Symposium on Computing at Minority Institutions (ADMI)*, Virginia Beach, 2017.

[8] L.-J. Z. Cheng Li, "A Blockchain-Based New Secure Multi-Layer Network Model for Internet of Things," in *Internet of Things (ICIOT), IEEE International Congress*, Honolulu, 2017.

[9] S. S. K. R. J. P. G. A. Dorri, "Blockchain for IoT security and privacy: The case study of a smart home," in *IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, Kona, Hawaii, 2017.

[10] M. M. S. T. R. B. S. Tuli, "FogBus: A Blockchain-based Lightweight Framework for Edge and Fog Computing," *arXiv,* vol. preprint , p. arXiv:1811.11978, 2017.

[11] J. L. K. R. C. M. Banerjee, "A blockchain future for internet of things security: a position paper," *Digital Communications and Networks ,* vol. 4, no. 3, pp. 149-160, 2018.

[12] S. S. K. a. R. J. A. Dorri, "Blockchain in internet of things: Challenges and Solutions," *arXiv,* p. arXiv:1608.05187, 2016.

[13] S. S. K. a. R. J. A. Dorri, "Towards an optimized blockchain for IoT," in *IEEE/ACM International Conference on Internet-of-Things Design and Implementation (IoTDI)*, Pittsburgh, 2017.

[14] S. S. K. R. J. P. G. A. Dorri, "LSB: A Lightweight Scalable BlockChain for IoT Security and Privacy," *arXiv,* p. arXiv:1712.02969, 2017.

[15] O. Novo, "Blockchain Meets IoT: An Architecture for Scalable Access Management in IoT," *IEEE Internet of Things Journal,* vol. 5, no. 2, pp. 1184-1195, 2018.

[16] Y. J. J. C. e. a. Y. Qian, "Towards decentralized IoT security enhancement: A blockchain approach," *Computers & Electrical Engineering,* vol. 72, pp. 266-273, 2018.

[17] O. A. e. al., "IoTChain: A blockchain security architecture for the Internet of Things," in *2018 IEEE Wireless Communications and Networking Conference (WCNC)*, Barcelona, 2018.

[18] L. W. Y. S. a. P. L. L. Zhou, "BeeKeeper: A Blockchain-Based IoT System With Secure Storage and Homomorphic Computation," *IEEE Access,* vol. 6, pp. 43472-43488, 2018.

[19] M. M. J. C. O. Ron Ross, "Systems Security Engineering: Considerations for a Multidisciplinary Approach in the Engineering of Trustworthy Secure Systems," *NIST Special Publication ,* Vols. 800-160, p. 230, 2016.

[20] E. A. C. T. Jie Li, "A General SDN Based IoT Framework with NVF Implementation," *ZTE Communication ,* vol. 13, no. 3, 2015.