Blockchain and Federated Learning-enabled Distributed Secure and Privacy-preserving Computing Architecture for IoT Network

1st Pradip Kumar Sharma
Department of Computing Science
University of Aberdeen
Aberdeen, UK
pradip.sharma@abdn.ac.uk

2nd Prosanta Gope
Department of Computer Science
University of Sheffield
Sheffield, UK
p.gope@sheffield.ac.uk

3rd Deepak Puthal
Department of Computer Science
Khalifa University
Abu Dhabi, UAE
deepak.puthal@ieee.org

***Abstract*—With the adoption of the 5G network, the exponential increase in the volume of data generated by the Internet of Things (IoT) devices, pushes the system to learn the model locally to support real-time applications. However, it also raises concerns about the security and privacy of local nodes and users. In addition, the approach such as collaborative learning where local nodes participate in the learning process of global model also raise critical concern regarding the cyber resilience of the network architecture. To address these issues, in this article, we identify the research gaps and propose a blockchain and federated learning-enabled distributed secure and privacy-preserving computing architecture for IoT network. The proposed model introduces the lightweight authentication and model training algorithms to build secure and robust system. The proposed model also addresses the reward and penalty issues of the collaborative learning with local nodes and propose a reward system scheme. We conduct the experimental analysis of the proposed model based on various parametric metrics to assess the effectiveness of the model. The experimental result shows that the proposed model is effective and capable of providing a cyber-resilience system.**

***Index Terms*—Blockchain, Federated Learning, Cyber Security, Security and Privacy, Internet of Things**

1. Introduction

With the adoption of network technology such as 5G, the legacy network architecture molds towards to accelerate to support real-time applications such as autonomous vehicles, industrial automation, augmented reality (AR) and virtual reality (VR), massive wearable devices, etc. [1] [2]. Due to the massive growth in volume of data generated by Internet of Things (IoT) devices around the world, traditional cloud-based network architecture is not sufficiently capable of delivering the features of 5G and support real-time applications [3] [4]. Edge computing network architecture has proven to be effective approach to address the limitations imposed by traditional cloud-based networks [5] [6]. With the increasing computing power of IoT devices, the model could run locally on edge server or gateway, or even on the local node itself, and the ability to process and store data faster for real-time applications that are essential for businesses. However, the 5G network also raises concerns about the security and privacy of local nodes and users. The network enables faster data transfer from large number of heterogeneous IoT devices to mobile operators with lower latency and higher bandwidth compared to the legacy network. The network will have to ensure that the organization or the intermediate servers are transparent with the way the data generated at the local nodes is processed and used in real time [7] [8]. Recently, the concept of federated learning makes it possible to preserve the privacy of sensitive data of local nodes and train the global model using collaborative learning by sharing only the parameters of the local model [9] [10]. In some cases, user's data at the edge network is limited by various policies and regulations and cannot be shared due to a privacy breach. The federated learning approach facilitates the learning of the global model without even recognizing and sharing the specific data of each user. However, federated learning also raises potential concerns such as a model poisoning attack, secure aggregation, a malicious local node participant, free riding attack, etc. [11] [12] [13] [14]. The integration of blockchain technology and federated learning has recently attracted a lot of attention to address the aforementioned problems. Many researchers are embracing the unaltered features of blockchain, federated learning, and differential privacy to secure the IoT network [15] [16] [17]. State-of-the-art methods also designed the incentive mechanism to reward local nodes and encourage participation in the learning process. However, there are issues such as lightweight authentication of the local participant node, reward and penalty system, the auditability of the successful participation of local model updates, etc. which are untouched and open. In addition, cyber resilience is another critical concern that we should take into account when designing a secure distributed architecture for a scalable IoT network. In this article, we propose a distributed, secure, and privacy preserving-oriented computing architecture for IoT network by leveraging the strength of blockchain technology and federated learning. In the proposed model, we design a learning approach to train the global model securely using lightweight authentication scheme. We also present reward system to encourage participation of local node in the learning process and make the system more secure and robust. We summarize the research contribution of this article as follows:

- Based on our literature review, we identify the research gaps and present the open issues and design requirements to build a secure and robust distributed computing architecture for IoT network.

- We propose a blockchain and federated learning-enabled distributed secure and privacy-preserving computing architecture for IoT network. The proposed model introduces the lightweight authentication and model training algorithms to build secure and robust system.
- We also propose a reward system to reward and penalty the participating local node based on its accuracy of local model updates. The system introduce feedback-based reward policy to facilitate the global model to achieve high accuracy.
- To evaluate the feasibility of the proposed system and support the proof-of-concept, we perform the experimental analysis based on various parametric metrics. The experimental result shows that the proposed system outperform compared to base model and capable to mitigating cyber-attacks.

The structure of the article is as follows: In Section II, we discuss the approach of federated learning, state-of-the-art methods, and identify the research gaps; We discuss the proposed model, authentication scheme, model learning process, and reward system in Section III; Section IV presents the system analysis of the proposed model; In Section V, we present the experimental analysis and results of the proposed model; Finally, Section VI concludes the article.

2. Preliminaries

Federated learning allows the system to perform processing on the device without sharing its dataset with an intermediate node / server. However, it has many limitations due to its centralized nature model. On the other hand, the features of Blockchain technology complement the limitation of federated learning to design a secure distributed computing architecture that preserves privacy of the IoT network. In this section, we discuss the concept of federated learning, related works in blockchain-enabled federated learning, and open issues.

2.1. Federated learning

To address key challenges such as data exists in fragments at the local node and privacy leaks while sharing local data, the federated learning ensures collaborative learning preserving privacy by performing a device-centric approach. In the federated learning approach, each local node learns the model on its own device and shares model updates instead of learning after aggregating data on the server. The federated learning aggregator node collects model updates from each local node and combines all models using the federated averaging algorithm. The aggregator node sends the aggregated model to each local node and the process will repeat until the aggregated model achieves optimal accuracy. Suppose there are local node nodes selected to participate in the model learning process with their own local datasets. The aggregate model in federated learning can be defined as follows [18]:

$$\omega^{t+1} = \frac{1}{N} \sum_{n \in M_t} \omega_k^{t+1} \tag{1}$$

Where M_t is subset of local nodes N, ω_n^{t+1} is update received from local nodes.

2.2. Related works

Nagar, A. [19] presents how the distributed learning architecture preserves privacy by integrating blockchain technology and federated learning with differential data sharing. They discussed the idea of adopting a federated learning ecosystem with existing technologies to support integration with Blockchain. To build a federated learning architecture without an aggregator, Ramanan, P. et al. [20] studied the potential characteristics of Blockchain technology. They argued that the operational and computational advantages of federated learning without an aggregator have significant potential for solving IoT network issues. To alleviate the challenges in fog computing such as local autonomy, latency and network congestion, a bloclachain-based federated leaning system is proposed by Qu, Y [21]. In the proposed system, the fog servers will generate and store the global model uploaded from each local devices. They presented the access control, verification and identity generation approaches in the proposed system to allow decentralized protection of privacy while preventing the failure of a single point. Hieu, N. Q. et al. [22] addresses the latency of training issue in the federated learning system enabled by the blockchain and presented a deep reinforcement-learning scheme for resource management. They formulated the stochastic optimization problem for managing the resources of the owner of the machine-learning model in order to minimize training latency and energy consumption. Pokhrel, S. R. and Choi, J. [23] proposed a design for autonomous blockchain-enabled federated learning in the vehicular network. They assigned each autonomous vehicle as a local node where the local model will train and will be exchanged and verified in a distributed manner. For the home appliance manufacturer industries, a reputation-based federated learning system is designed to help manufacturers develop a smart home system [24]. The proposed system can be used to train model in order to predict the needs and consumption behavior of customers in the future. They integrated federated learning, blockchain and mobile computing to create a secure decentralized system with differential confidentiality to protect the confidentiality of customer data. Lu, Y. et al [25] presented a secure data sharing architecture using blockchain and federated learning. They formulated the problem of sharing data in the distributed system by incorporating federated learning approach to reduce the risk of data leakage. Preuveneers, D et al. [26] conducted a case study on a machine learning model for anomaly detection using blockchain and federated learning. They argued that nodes participating in federated learning could be held responsible for auditing model updates.

2.3. Open issues and design requirements

Based on the above literature review, we have identified unresolved issues and design requirements for the secure distributed computing architecture in the IoT network. The unresolved issues and design requirements are as follows:

- *Light weighted authentication scheme*: Local nodes collect sensitive environmental information and provide local model updates based on a local dataset. Updates from the local node will be used to train the global model, therefore the security solution to provide secure interaction should be lightweight due to resource constraints.
- *Auditable local model updates*: The system must ensure auditability so that any stakeholder can determine if the resulting local node updates have been manipulated. Because a malicious participant may try to inject fake local model updates to deliberately reduce the performance of the global model.
- *Feedback based reward system*: The feedback-based reward system indicates that the system will reward the local participating node based on its accuracy of local model updates during the learning process of the global model. The network architecture with a feedback-based reward policy facilitates the global model to achieve high accuracy and motivates local nodes to participate in the learning process. At the same time, the system should also penalize the local participant nodes if local model updates reach lower performance to avoid malicious participants, and other cyber-attacks.
- *Contribution-based rewards*: It can still be argued that the system should share the profit (i.e. rewards) equally at all participating local nodes. In some scenarios, this approach is unfair and arguable. Suppose there are two local participating nodes with their own dataset with a different data size. Even if the two model updates of the local node model achieve the same accuracy rate, the reward system must all take into account the size of each local node dataset. The local node with large data storage is more likely to upgrade the performance of the global model compared to another local node. In addition, maintaining a large dataset also increases the overall cost of the local node. Therefore, the system should consider contribution-based rewards when designing the computing network architecture.
- *Cyber resilience*: In the cyber world, cyber resilience in the architecture of computing networks is of utmost importance. The system should have the ability to prepare for, react to and recover from cyber-attacks such as data poisoning attacks, free riding attacks, malicious participants, etc.

3. System modeling

In this section, we present a new distributed, secure, and privacy preserving-oriented computing architecture based on Blockchain and federated learning for the IoT network. Subsequently, we define the secure authentication, model training and reward system modules of the proposed system.

3.1. System design

Fig. 1 illustrate the proposed blockchain and federated learning-enabled distributed secure and privacy-preserving computing architecture for IoT network. The proposed model compose of four modules: local nodes, edge nodes, blockchain-enabled distributed fog network, and core distributed cloud. Local nodes can be edge network devices such as smartphones, smart gateways, smart vehicles, etc., which have computing resources and storage capacity to participate in the model learning process. Each local node uses its own local dataset, storage and compute resources to participate in the model learning process. The edge node consist of set of miner nodes and is responsible for the secure aggregation of the global model. At edge node level, each time a new block is created to add the global train model to the distributed network, it is first validated from peer-edge nodes. Each edge node also maintains an individual off-chain to store the intermediate models while training global model in the iterative process. We use a distributed fog network using consortium blockchain for our crowdsourcing system to store global models permanently and the core distributed cloud is responsible for initial authentication and key distribution. We discuss the proposed model in detail in the following subsections.

3.2. Secure authentication

Each local node n_i consist of Physical Unclonable Function (PUF) ID to provide unique identity of physical object, device-unique data security and authentication solution for the cyber network. To register the local node into the distributed network to securely patriciate in the training process of the model, PUF symmetric key $k_i\,(s)$ is generated for each local node at the edge of the network. Once the local node symmetric key is generated, we use the generated symmetric key to derive public and private key pairs $(k_{pu},\ k_{pv})$ via True Random Number Generators (TRNGs).

In the PyPuf simulator, the uniqueness of two different PUFs responses measure using hamming distance [27]. The hamming distance of the responses of two PUFs can be calculate as:

$$Distance\,(PUF_{Ri},\ PUF_{Rj}) = \sum_{l-0}^{k} PUF_{Ria} \oplus PUF_{Rja}$$

Where PUF_{Ria} is the response of PUF_{Ri}. For each index, we can acquire a vector that contains zeroes where two *response vectors* were equal. We use the hamming distance of the responses two PUFs to calculate the uniqueness. The uniqueness function different PUFs responses of can be defined as:

$$Uniqueness\,(PUF_{Ri},\ PUF_{Rj}) = \frac{2}{m(m-1)}\sum\sum\frac{Distance\,(PUF_{Ri},\ PUF_{Rj})}{k}$$

Where, m is the arbiter physical unclonable functions chains, and n is number of bits.

To register the local node into the core distributed cloud (CDC), the node n_i initiate request Req_{n_i} :

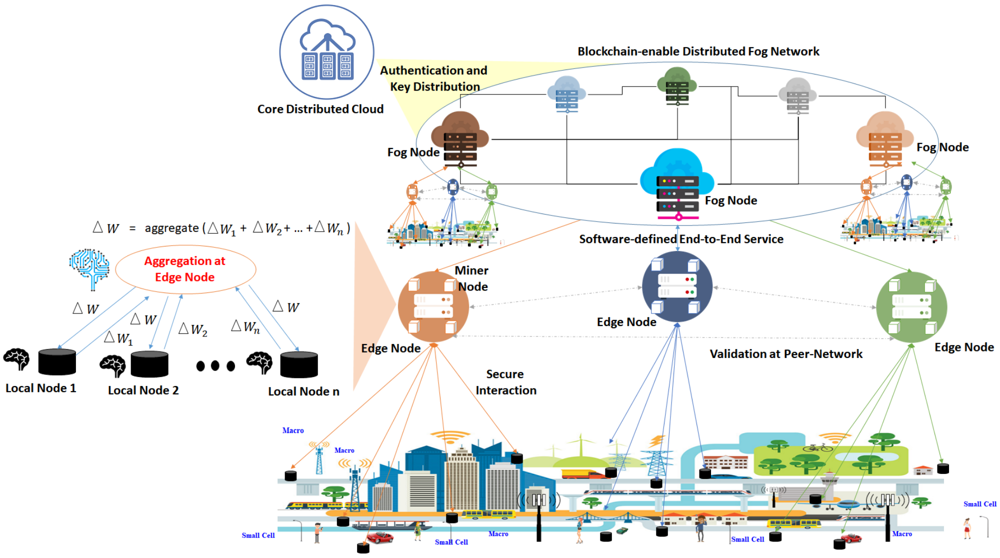


Figure 1. Proposed system architecture: Integrating blockchain and federated learning for distributed secure and privacy-preserving computing architecture

$\{k_{pu},\ h(k_{pu})\}$ through a secure channel. Here we leverage the strength of LAAP authentication protocol proposed by Gope [28]. Likewise in LAAP scheme, the CDC maintains a global counter to generate sequence number *Tseq,* shared key k_s, and a set of un-linkable pseudo IDs $PID = \{pid_1,\ pid_2,\ pid_3,\ \dots,\ pid_n\}$. The global counter is incremented by one each time the CDC receives a request from the local node. By keeping a copy in its database, the CDC provides these parameters to the local node n_i through secure channel. In addition, CDC generates a smart contract along with all parameters, including generated public key by the PUF ID of the local node, and is deployed in the blockchain enable distributed network. Note that the proposed model uses two different key pairs for communications: the key pairs generated by CDC is use for secure communications and data exchange; while the key pairs generated at local node using PUF ID is use for communication when participating in the training model.

3.3. Model training

To initiate the training model using federated leaning in distributed network, at $t = 0$, the distributed network initialize the parameters of global model based on the proxy data available at CDC. The CDC create a block consists of proxy global model, and its parameters and add into the blockchain network. As mentioned in previous section, each local node consist of two key pairs. The key pairs generated from PUF ID of each local node is use for secure aggregation during training model. First, local node sends request for participation to train the model based on their available local data. Let suppose the edge node receive N participants request from local nodes$\{p_{req1}, p_{req2}, p_{req3}, \dots, p_{reqn}\}$. To achieve optimal accuracy and prevent security attacks, edge node is calculate score of each requested local nodes and select M participants to train the model using federated learning scheme. The score of each local can be calculated as follows:

$$Weight_{p_{reqi}} = SD_i + \frac{1}{LT_i} + \frac{1}{Stack_i}$$

$$Score_{p_{reqi}} = \frac{N(N-1)}{2} \times Weight_{p_{reqi}}$$

Where N is the number of participants request from local nodes, $Stack_i$ is the rewards earn by each local node, and LT_irepresents the time at which local node participated last time to generate train model. Stack is use in the proposed model to reward the local node for successfully participate to train the model.SD_i represents the standard deviation of the local node participated last time to generate train model. SD of each local node is calculate using shared parameters based on local train model and aggregated parameters of global train model. It can be defined as follows:

$$SD_i = \left| y_i - \frac{\sum_{i-1}^{n} |y_i - \hat{y}_i|}{n} \right|$$

Second, to preserve the privacy of the each model of local node, the edge node generate noise for M selected participants and send to each local participant node by encrypting with their respected shared public key. In the proposed model, we use parallel composition differential privacy technique [29]. We consider the all the local dataset of M selected participants is a single dataset which is further partitioned into M disjoint subsets. If there are M independent mechanism $\{M_1,\ M_2,\ \dots,\ M_m\}$ with privacy guarantee $\{\epsilon_1, \epsilon_2,\ \dots, \epsilon_m\}$, the differential of the function $f\,(M_1,\ M_2,\ \dots,\ M_m)$ is $\sum_{i=1}^{M} \epsilon_i$. In case of parallel composition, the function $f$is $\max_i \epsilon_i$. Let suppose $Local_{DS1}$ and $Local_{DS2}$ are two local training datasets at local node LT_1 and LT_2 then parallel composition differential privacy theorem follows following property:

$$P_r\left[M\left(Local_{DS1}\right) = Subset_n\right] =$$

$$\prod_{l=1}^{m} P_r\left[M_l\left(Local_{DS2}\right); Subset_1, Subset_2, \dots,\right.$$

$$\text{Subset}_{l-1} = Subset_l]$$

$$\leq exp\left(\epsilon\right) P_r \left[M_k\left(Local_{DS2}\right); Subset_1, Subset_2, \ldots,\right.$$

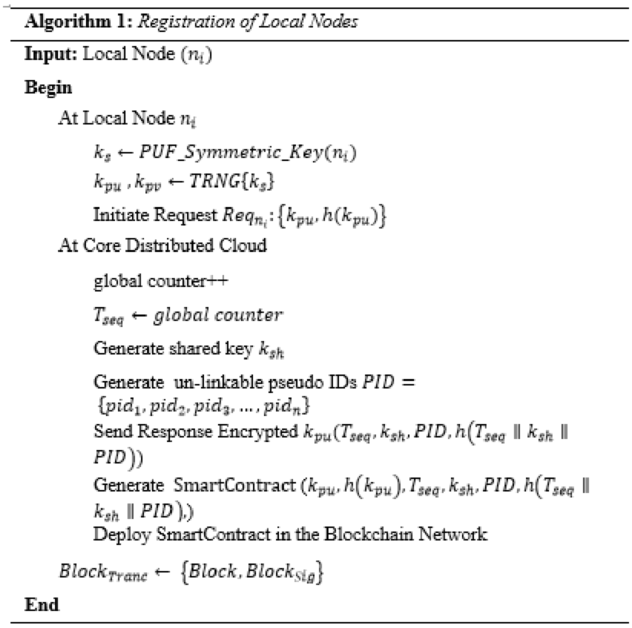
$$\left.\text{Subset}_{k-1} = Subset_k\right]$$

$$\prod_{l\neq k}^{m} P_r \left[M_k\left(Local_{DS1}\right); Subset_1, Subset_2, \ldots,\right.$$

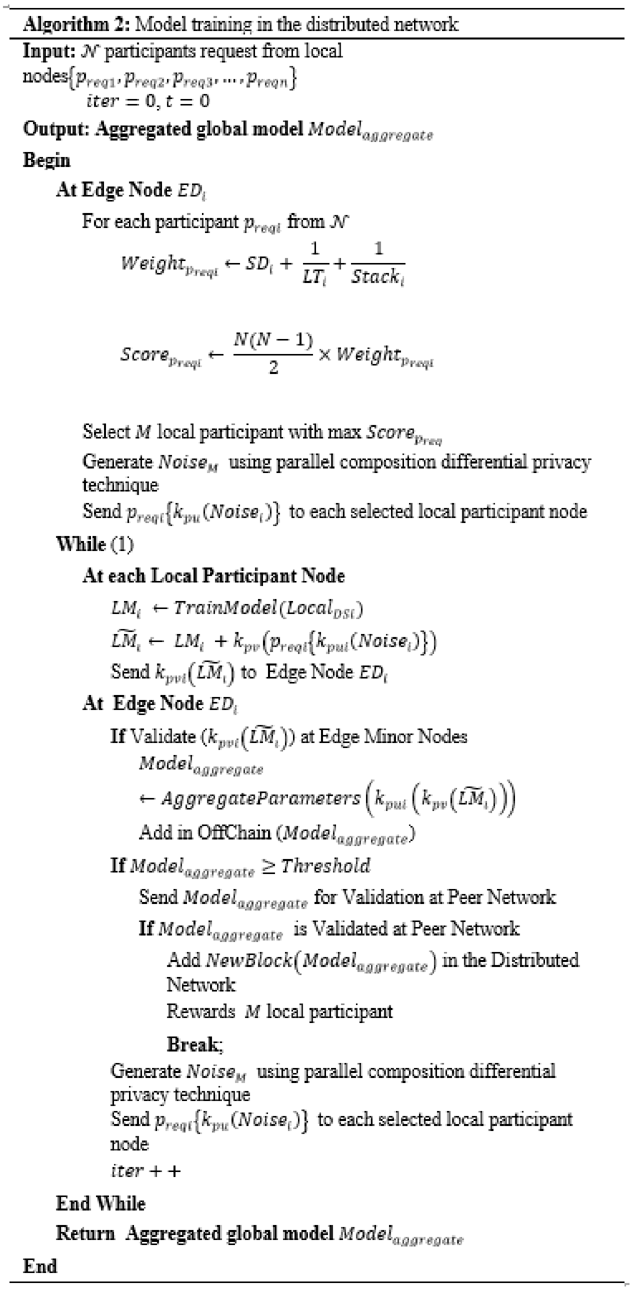
$$\left.\text{Subset}_{k-1} = Subset_k\right]$$

$$= exp\left(\epsilon\right) P_r \left[M\left(Local_{DS2}\right) = Subset_n\right]$$

Third, all of the selected participating nodes train the model locally based on their locally available datasets. In this phase, each selected participating node adds noise in the parameters of the locally trained model and shares with the edge node by encrypting with their own private key. Each shared local parameter is validated within the minor node at the edge node level and adds a new block with model parameters aggregated at the edge node off-chain. The edge node iterates the process to train the aggregate global model until the accuracy rate reaches the threshold value or the optimal rate. Once the aggregate global model achieves an optimal accuracy rate, the edge mode sends the newly created block with aggregated model parameters for validation on the peer network. If the block is validated on the peer network, the edge node sends the newly created block to add to the distributed blockchain network. Algorithm 2 presents each of the stages of model formation. In the proposed model, the edge node also rewards to each local node for their successfully participation in the model training process. We discuss the reward model in the next subsection.



```
Algorithm 1: Registration of Local Nodes
Input: Local Node ($n_i$)
Begin
    At Local Node $n_i$
        $k_s \leftarrow PUF\_Symmetric\_Key(n_i)$
        $k_{pu}, k_{pv} \leftarrow TRNG\{k_s\}$
        Initiate Request $Req_{n_i}: \{k_{pu}, h(k_{pu})\}$
    At Core Distributed Cloud
        global counter++
        $T_{seq} \leftarrow global\ counter$
        Generate shared key $k_{sh}$
        Generate un-linkable pseudo IDs $PID = \{pid_1, pid_2, pid_3, \ldots, pid_n\}$
        Send Response Encrypted $k_{pu}(T_{seq}, k_{sh}, PID, h(T_{seq} \parallel k_{sh} \parallel PID))$
        Generate SmartContract $(k_{pu}, h(k_{pu}), T_{seq}, k_{sh}, PID, h(T_{seq} \parallel k_{sh} \parallel PID),)$
        Deploy SmartContract in the Blockchain Network
    $Block_{Tranc} \leftarrow \{Block, Block_{Sig}\}$
End
```



```
Algorithm 2: Model training in the distributed network
Input: $\mathcal{N}$ participants request from local nodes$\{p_{req1}, p_{req2}, p_{req3}, \ldots, p_{reqn}\}$
        $iter = 0, t = 0$
Output: Aggregated global model $Model_{aggregate}$
Begin
    At Edge Node $ED_i$
        For each participant $p_{reqi}$ from $\mathcal{N}$
            $Weight_{p_{reqi}} \leftarrow SD_i + \frac{1}{LT_i} + \frac{1}{Stack_i}$

            $Score_{p_{reqi}} \leftarrow \frac{N(N-1)}{2} \times Weight_{p_{reqi}}$

        Select $M$ local participant with max $Score_{p_{req}}$
        Generate $Noise_M$ using parallel composition differential privacy technique
        Send $p_{reqi}\{k_{pu}(Noise_i)\}$ to each selected local participant node
    While (1)
        At each Local Participant Node
            $LM_i \leftarrow TrainModel(Local_{DSi})$
            $\widetilde{LM}_i \leftarrow LM_i + k_{pv}(p_{reqi}\{k_{pui}(Noise_i)\})$
            Send $k_{pvi}(\widetilde{LM}_i)$ to Edge Node $ED_i$
        At Edge Node $ED_i$
            If Validate $(k_{pvi}(\widetilde{LM}_i))$ at Edge Minor Nodes
                $Model_{aggregate}$
                $\leftarrow AggregateParameters\left(k_{pui}\left(k_{pv}(\widetilde{LM}_i)\right)\right)$
                Add in OffChain $(Model_{aggregate})$
            If $Model_{aggregate} \geq Threshold$
                Send $Model_{aggregate}$ for Validation at Peer Network
                If $Model_{aggregate}$ is Validated at Peer Network
                    Add $NewBlock(Model_{aggregate})$ in the Distributed Network
                    Rewards $M$ local participant
                    Break;
            Generate $Noise_M$ using parallel composition differential privacy technique
            Send $p_{reqi}\{k_{pu}(Noise_i)\}$ to each selected local participant node
            $iter++$
    End While
    Return Aggregated global model $Model_{aggregate}$
End
```

3.4. Reward system

In the proposed model, $Stack_i$ represents the rewards earn by each local node (n_i). In the reward system, the system assign stack to local node for each successfully participation in the model training process. A reward can be positive or negative in the proposed system to deal with the cyber-attacks and make system more robust. In case of negative reward, the system corresponds to penalty during the model training process. The system calculates the reward for each local node participant as follows:

$$Reward_i =$$

$$\text{Size}(Local_{DSi})\frac{}{Count(p_{req1}, p_{req2}, p_{req3}, \ldots, p_{reqn}) \times Reward_{Pool}//}$$
$$+\ \vartheta\sigma_i$$

$$\sigma_i = \left| y_i - \frac{1}{m}\sum_{j=0}^{m} y_j \right|$$

$$\vartheta = \left\{ \begin{array}{c} 1If \;\; |y_i - y| \leq threshold\_value \\ -1Else \end{array} \right.$$

Where, $Size\left(Local_{DSi}\right)$ represents the size of the local dataset of node (n_i); $Reward_{Pool}$ represents the reward pool for each successful training model; $threshold\_value$ is the value set for penalty. $Reward_{Pool}$ is also depend on the rate of accuracy achieve by the global model and can be calculated as follows:

$$Reward_{Pool} = ActualReward_{Pool} \times Rate\_of\_Accuracy$$

Finally, the stack of each local node update as follows: $Stack_i = Stack_i + Reward_i$. The size of the local dataset is one of the crucial factors that we have taken into account in the reward system. The node with a high volume of local data is more likely to contribute during the learning model to achieve high accuracy of the global model.

4. System analysis

4.1. Secure and Robustness

- *Secure authentication*: The secure authentication and key exchange are the important issues for the security of the distributed computing network for IIoT. One of the very first security requirements for IoT devices is authentication, that is to say making sure the unit is genuine. The proposed model to achieve the auditability, security and the privacy preserving during constructing transactions. We use the PUF ID in the proposed model, which relies on the unpredictability of its response for a given challenge based on complex interactions with a physical function. Authentication of IoT device using PUF ID is secure if for the two input challenges $Challenge_1$, $Challenge_2 \in (0,1)^n$ generate two responses $Response_1$, $Response_2 \in (0,1)^n$ with at least k variations. Mathematically, the authentication of IoT device using PUF ID is secure it satisfies the following properties:

  $$P_r\left[Distance\left(PUF_i\left(Challenge_1\right),\right.\right.$$
  $$\left.\left.\text{PUF}_i\left(Challenge_2\right) > k = 1 - \tau\right.\right.$$
  $$P_r\left[Distance\left(PUF_i\left(Challenge_1\right),\right.\right.$$
  $$\left.\left.\text{PUF}_j\left(Challenge_1\right) > l = 1 - \tau\right.\right.$$

  Where k and l are the error tolerance thresholds, τ is a negligibly small value, and $Distance$ represents the hamming distance.
- *Decentralized privacy protection*: By integrating blockchain, federated learning, and differential privacy in the distributed IIoT network, the proposed model allows decentralized privacy protection and prevents single point failure. As algorithm 2 shows, with each iteration of processing of model training, the noise is added to the local model using a differential privacy technique.
- *Decentralized trust and structurally scalable*: The potential features of blockchain technology in the proposed model remove the high risk of data leakage, which is very likely to incur in centralized trust. In addition, the federated learning feature to share the parameters of the locally train model instead of sharing local data improves trust of the proposed model. It also facilitates the architecture structurally more scalable by requiring only the exchange of training updates.
- *Secure aggregation* : At single point entity, aggregation at a single node is likely to compromise during a cyber-attack. If the single-node aggregator is compromised, the attacker can easily control and poison global model. In the proposed model, the edge node consists of set of minor nodes to aggregate the update of the local learning model received from the local nodes.
- *Incentive and profit sharing*: As discussed in the previous section, the reward will be given to each participant in the proposed model who has successfully participated in the training of the global model. Rewards will be distributed to each participating node based on the scores of their submitted updates.

4.2. Security Attacks

- *Data poisoning attack*: In federated learning, the resource constrained local node train the model locally and share model updates while retaining local training data can allow data privacy and security. However, the local node could be exploited by attackers and inject poisoned data to degrade the performance of the global model. In the proposed model, each local node updates is validated with minor nodes at the edge node. If the local node updates is drastically different from other local node updates, the minor nodes at edge node will not approve model updates. Let suppose if the attackers compromise the multiple local nodes, it may difficult to identify the data poisoning attack at edge node. To address this scenario, when the edge node create a new block, it must send for validation to peer network, as indicated in Algorithm 2. The new global model will not update in the distributed network until the peer network approves it. If the newly created block is rejected from peer network, the model will drop from the off-chain at the edge node and impose a penalty on each participating local node using the reward system function.
- *Free riding attack*: Each local node participant in federated learning should contribute to train the global model to achieve a high accuracy rate. There may be a scenario in which each participating node could get a reward by pretending to contributing to train global model. In free riding attacks, the participating node submits fake updates for a reward, either it does not have enough data (or cares about data privacy), or it may want to save local computing resources. In the proposed model, the reward system function will only reward the participating node when the accuracy rate of the individual node meets the reward criteria.

If the participating node attempts to launch free riding attacks by providing false updates, a penalty will be imposed.

- *Malicious participant attack*: Sometimes, a malicious participant node submits incorrect information to poison the global model and gets lower performance. In the proposed model, the participant node will be selected to train global model based on $Score_{p_{reqi}}$ of each node requested. The model calculate the $Score_{p_{reqi}}$ according to the weight function defined in the previous section. To deal with the malicious participant attack, the proposed model will allow a list of limited node to participate in the training phase of the global model.
- *Model poisoning attack*: The main challenges of the model poisoning attack are how to secure the local models sent from the local node to the edge node and how to prevent the model from being exposed to unauthorized devices and from poisoning. In the proposed model, each local model updates will be send by encrypting with its own private key generated from their individual PUD ID. As defined in Algorithm 1, each local node will keep two pairs of keys: one is use for secure authentication and data exchange; and the other is use to participate in the training phase of global model.

5. Experimental Analysis

5.1. Experimental Setup

In this section, we assess the proposed model to show the effectiveness and how the model is efficient enough to provide secure and robust system. We perform the evaluation of our model on a real world dataset [30]. To simulate the proposed model prototype, we install Jupyter notebook, Python 3.7, and Tensorflow 2.0. We consider the various attack scenarios such as free riding attack, data poisoning attack, malicious participant, etc. to assess the effectiveness of the model. To perform the activities of minor nodes at the local node level, we simulate the authorization Blockchain on top of a private Ethereum blockchain network. To configure the local edge node, we used go-ethereum and installed Mist. We discuss the different experimental scenarios and the results in the following subsection.

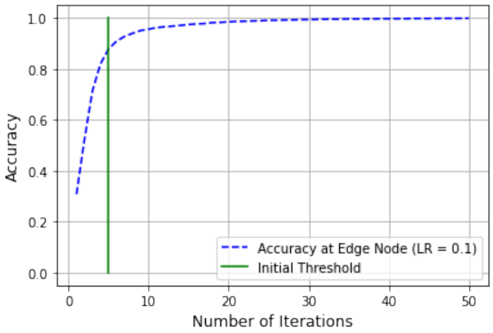


Figure 2. Rate of accuracy achieved by the proposed model at edge node

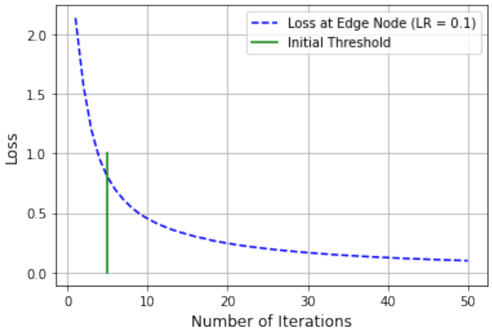


Figure 3. Rate of loss of the aggregated model at LR 0.01

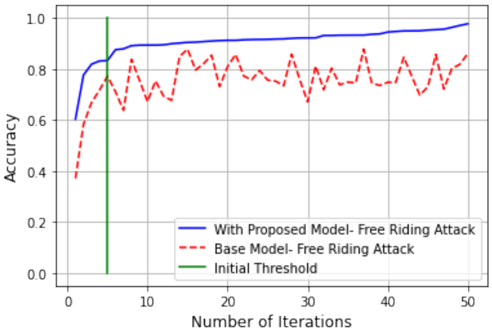


Figure 4. Accuracy rate during free riding attack

5.2. Experimental Results

Initially, we created 100 different local nodes and randomly partition the dataset among 100 local nodes. Also, we generate a proxy dataset based on the dataset available at local nodes. In the first scenario, we trained the global model at edge node by considering all the local nodes with their own local dataset in attack-free environment. We set the value of hyper leaning rate (LR) at 0.01 and observed the accuracy and loss of the aggregated model at edge node. Fig. 2 shows the result of accuracy rate achieved by the proposed model at edge node. The results indicate that the aggregated model achieved 0.986 accuracy rate in 50 iterations (we repeated the steps 10 times to check the precision of the result). We also present the observed rate of loss of the aggregated model at LR 0.01 in Fig. 3. As shown in the Fig. 2 and Fig. 3, based on the size and nature of the dataset at local nodes, we set the initial threshold value 5 in our experimental analysis. This is a critical parameter that we need to take into account due to the distributed and heterogeneous nature of the dataset at the local node level in real-time application scenarios.

In the second scenario, we assess the effectiveness of the proposed model in free riding attack scenario. We randomly select 20 clients and generate values of the gradient updates matrix to launch free riding attack. We defined 20 local nodes as clients and assign generated gradient updates matrix to pretend that they have their own local dataset that could be used to train the global model. We observed the accuracy and loss of the aggregated model at edge node in free riding attack scenario with LR at 0.01 and initial threshold at 5. Fig. 4 shows the result of accuracy rate achieved by the proposed model and base model during free riding attack. As we can see in Fig. 4, in the base model the accuracy rate fluctuates with each iteration due to fake updates and achieved a low rate of accuracy. While in the proposed model, using the proposed Algorithm 2, the model can mitigate the free riding attack and achieve a high stable accuracy rate with each iterations. We also observed the rate of loss of the aggregated model in Fig. 5.

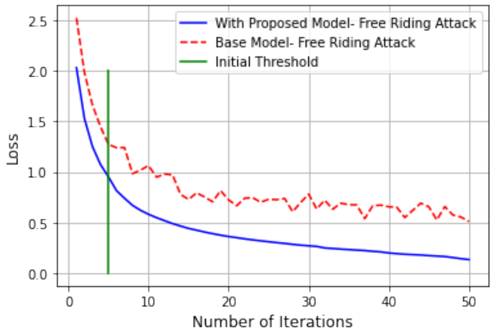


Figure 5. Rate of loss during free riding attack

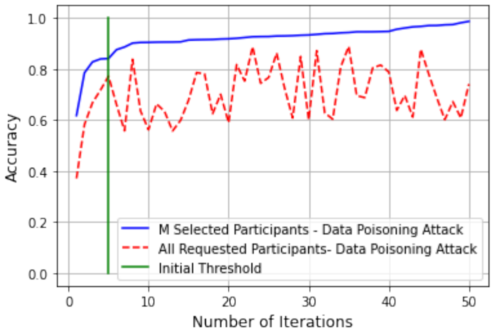


Figure 6. Response of proposed model during data poisoning attack

We considered the data poisoning attacks and malicious participants in third scenario. In this scenario, attacker participate in the training process to submits incorrect information to poison the global model and gets lower performance. In our experimental analysis, we assigned some random clients as malicious participants and injected poisoned data to launch data poisoning attack. We observed the behavior of the proposed model in terms of accuracy rate. We considered two different cases: first, we selected all the requested local nodes to participate in the process of training global model; and secondly, we selected M participants from all requested local nodes $\{p_{req1}, p_{req2}, p_{req3}, \ldots, p_{reqn}\}$ based on the calculated score of each local requested node. Fig. 6 shows the result of accuracy rate achieved in both cases. As shown in Fig. 6, in the first case, where all requested local nodes participated in the training process, the accuracy rate highly fluctuates with each iteration and attacker has successfully lowered the performance of the model. Whereas, in the second case where the proposed model selected M participants from all requested local nodes, succeeded in mitigating the data poisoning attacks without degrading the performance of the model.

To evaluate the reward system in the proposed model, we also observed the score received all local node for successful participation in the model training process. As we discussed earlier, the reward score of each local node can be positive or negative based on their contribution in the model training process. Fig. 7 shows the reward score received by each local node who participated successfully. In the experimental analysis, we assigned the value of ϑ 0.10 to calculate the reward and penalty of each local node. As shown in Fig. 7, some local nodes have obtained a negative reward score calculated based on the proposed reward scheme. This results in ensuring the proposed model to improve the performance of the global model and to mitigate the system against cyberattacks such as free riding, data poisoning, and malicious participants, etc.

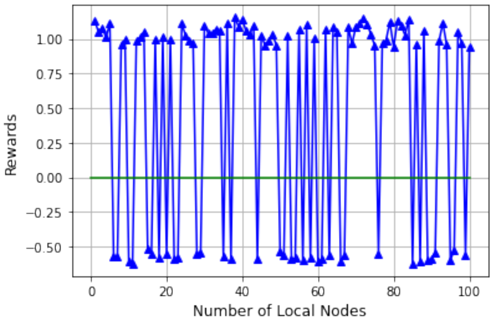


Figure 7. Reward score received by each local node who participated successfully in learning process

Experimental results show that the efficiency of the proposed model and the reward system allow the system to be safer and more robust. Furthermore, the global model updates are validated and stored in the distributed Blockchain network rather than on a single central server, which makes the system more secure, transparent, and robust compared to the legacy network architecture.

6. Conclusion

The integration of the blockchain technology and collaboration learning approach such as federated learning has attracted considerable attention in recent years and is a promising a way to build secure and robust computing architecture for IoT network. But, cyber resilience becomes a critical obstacle and needs to addressed to realize the system capable of handling real-world application scenarios. In this work, we proposed distributed computing architecture using the features of blockchain and federated learning to build secure and robust system. The model introduced an efficient lightweight authentication scheme and learning methods to train the global model by participating the local nodes in the learning process. The model also proposed reward system to reward and penalize the participating nodes in order to obtain high performance of the global model. The experimental results are promising and effective in mitigating cyber-attacks.

References

[1] D. C.Nguyen, P. N. Pathirana, M.Ding, and A. Seneviratne, ”Blockchain for 5G and beyond networks: A state of the art survey,” Journal of Network and Computer Applications, vol. 166, pp. 102693, 2020

[2] S. Li, L.Da Xu, and S. Zhao, ”5G Internet of Things: A survey. Journal of Industrial Information Integration, vol. 10, pp. 1-9, 2018

[3] J. Cao, M. Ma, H.Li,R. Ma, Y. Sun, P. Yu, and L. Xiong, ”A Survey on Security Aspects for 3GPP 5G Networks,” IEEE Communications Surveys & Tutorials, vol. 22, pp. 170-195, 2019

[4] Y. Wu, A. Khisti, C. Xiao, G. Caire, K. K. Wong, and X. Gao, ”A survey of physical layer security techniques for 5G wireless networks and challenges ahead,” IEEE Journal on Selected Areas in Communications, vol. 36, pp. 679-695, 2018

[5] A. Yousefpour, C. Fung, T. Nguyen, K. Kadiyala, F. Jalali, A. Niakanlahiji, ... and J. P. Jue, ”All one needs to know about fog computing and related edge computing paradigms: A complete survey,” Journal of Systems Architecture, vol. 98, pp. 289-330, 2019

[6] W. Z. Khan, E. Ahmed, S. Hakak, I. Yaqoob, and A. Ahmed, ”Edge computing: A survey,” Future Generation Computer Systems, vol. 97, pp. 219-235, 2019

[7] R. Roman, J. Lopez, and M. Mambo, ”Mobile edge computing, fog et al.: A survey and analysis of security threats and challenges,” Future Generation Computer Systems, vol. 78, pp. 680-698, 2018

[8] J. Zhang, B. Chen, Y. Zhao, X. Cheng, and F. Hu, ”Data security and privacy-preserving in edge computing paradigm: Survey and open issues,” IEEE Access, vol. 6, pp. 18209-18237, 2018

[9] Q.Yang, Y.Liu, T. Chen, and Y. Tong, ”Federated machine learning: Concept and applications,” ACM Transactions on Intelligent Systems and Technology (TIST), vol. 10, pp. 1-19, 2019

[10] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, and K. Chan, ”Adaptive federated learning in resource constrained edge computing systems,” IEEE Journal on Selected Areas in Communications, vol. 37, pp. 1205-1221, 2019

[11] A. N.Bhagoji, S. Chakraborty, P. Mittal, and S. Calo, ”Model poisoning attacks in federated learning. In In Workshop on Security in Machine Learning (SecML),” collocated with the 32nd Conference on Neural Information Processing Systems (NeurIPS’18),(2019, December)

[12] D. Cao, S. Chang, Z. Lin, G. Liu, and D. Sun, ”Understanding distributed poisoning attack in federated learning,” In 2019 IEEE 25th International Conference on Parallel and Distributed Systems (ICPADS) (pp. 233-239). IEEE, (2019, December)

[13] C.Fung, C. J.Yoon, and I. Beschastnikh, ”Mitigating sybils in federated learning poisoning,” arXiv preprint arXiv:1808.04866, 2018

[14] L. Su, ”Defending Distributed Systems Against Adversarial Attacks: Consensus, Consensusbased Learning, and Statistical Learning,” ACM SIGMETRICS Performance Evaluation Review, vol. 47, pp. 24-27, 2020

[15] H. Kim, J. Park, M. Bennis, and S. L. Kim, ”Blockchained on-device federated learning,” IEEE Communications Letters, vol. 24, no. 6, pp. 1279-1283, 2019

[16] J. Weng, J. Weng, J. Zhang, M. Li, Y. Zhang, and W. Luo, ”Deepchain: Auditable and privacy-preserving deep learning with blockchain-based incentive,” IEEE Transactions on Dependable and Secure Computing, vol. 18, no. 5, pp. 2438-2455, 2019

[17] J. Kang, Z. Xiong, D. Niyato, Y. Zou, Y. Zhang, and M. Guizani, ”Reliable federated learning for mobile networks,” IEEE Wireless Communications, vol. 27, pp. 72-80, 2020

[18] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, ”Federated optimization in heterogeneous networks,” arXiv preprint arXiv:1812.06127, 2018

[19] A. Nagar, ”Privacy-Preserving Blockchain Based Federated Learning with Differential Data Sharing,” arXiv preprint arXiv:1912.04859, 2019

[20] P. Ramanan, K. Nakayama, and R. Sharma, ”BAFFLE: Blockchain based Aggregator Free Federated Learning,” arXiv preprint arXiv:1909.07452, 2019

[21] Y. Qu, L. Gao, T. H. Luan, Y. Xiang, S. Yu, B. Li, and G. Zheng, ”Decentralized Privacy using Blockchain-Enabled Federated Learning in Fog Computing,” IEEE Internet of Things Journal, vol. 7, no. 6, pp. 5171-5183, 2020

[22] N. Quang Hieu, N. Cong Luong, D. Niyato, D. In Kim, and E. Elmroth, ”Resource Management for Blockchain-enabled Federated Learning: A Deep Reinforcement Learning Approach,” *arXiv*, arXiv-2004, 2020

[23] S. R. Pokhrel, and J. Choi, ”Federated learning with blockchain for autonomous vehicles: Analysis and design challenges,” IEEE Transactions on Communications, vol. 68, no. 8, pp. 4734-4746, 2020

[24] Y. Zhao, J. Zhao, L. Jiang, R. Tan, and D. Niyato, ”Mobile edge computing, blockchain and reputation-based crowdsourcing iot federated learning: A secure, decentralized and privacy-preserving system,” *arXiv preprint arXiv:1906.10893*, 2019

[25] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, ”Blockchain and Federated Learning for Privacy-preserved Data Sharing in Industrial IoT,” IEEE Transactions on Industrial Informatics, 2019

[26] D.Preuveneers, V. Rimmer, I.Tsingenopoulos, J. Spooren, W. Joosen, and E. Ilie-Zudor, ”Chained anomaly detection models for federated learning: An intrusion detection case study,”Applied Sciences, vol. 8, pp. 2663, 2018

[27] O. Näslund, Lightweight and Machine Learning Attack Resistant Physical Unclonable Functions, 2019

[28] P. Gope, ”LAAP: Lightweight anonymous authentication protocol for D2D-Aided fog computing paradigm,” computers & security, vol. 86, pp. 223-237, 2019

[29] M. U. Hassan, M. H. Rehmani, and J. Chen, ”Differential privacy techniques for cyber physical systems: a survey,” IEEE Communications Surveys & Tutorials, vol. 22, no. 1, pp. 746-789, 2019

[30] P. K. Sharma, J. H. Park, and K. Cho, ”Blockchain and Federated Learning-based Distributed Computing Defence Framework for Sustainable Society,” Sustainable Cities and Society, vol. 59, pp. 102220, 2020