IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

An Integrated Federated Machine Learning and Blockchain Framework with Optimal Miner Selection for Reliable DDOS Attack Detection

D.Saveetha[1], G.Maragatham[2], Vijayakumar Ponnusamy[3] ,IEEE Senior Member, and Nemanja Zdravković[4]

[1]Department of Networking and Communications, SRM Institute of Science and Technology, Kattankulathur 603203, Tamil Nadu, India
[2]Department of Computational Intelligence, SRM Institute of Science and Technology, Kattankulathur 603203, Tamil Nadu, India
[3]Department of Electronics and Communication Engineering, SRM Institute of Science and Technology, Kattankulathur 603203, Tamil Nadu, India
[4]Faculty of Information Technology, Metropolitan University, Serbia

Corresponding author: Vijayakumar. P (e-mail: vijayakp@srmist.edu.in).

ABSTRACT Blockchain networks serve as a transparent and secure ledger storage solution, yet they remain vulnerable to attacks. There must be some mechanism to protect the blockchain network from attacks. Among various attacks, the Distributed Denial of Service (DDoS) attack is considered severe, which is challenging to detect accurately and reliably. Machine learning techniques are used to detect the attack, which requires exploring all global attack data in a single system, which is difficult in practice. This article proposes a distributed machine learning mechanism called Federated Machine Learning for detecting the presence of DDoS attacks. But in federated machine learning the model itself can be poisoned by the malicious collaborating node which is another problem that this article solves by storing the model in blockchain and by introducing a new reputation-based miner selection procedure. The proposed framework integrates the federation of machine learning within the blockchain network framework for detecting DDoS attacks. Under the integrated framework, miners are used to train the blocks and they also participate in the machine learning training. A dynamic reputation-based miner selection mechanism that can balance exploration and exploitation is proposed for optimal miner selection, which can ensure the high accuracy of the machine learning model and improve the security of blockchain from attacks like DDoS attacks and 51% attacks. The proposed framework is tested with Random Forest, Multilayer Perceptron, and Logistic Regression machine learning algorithms. The proposed mechanism achieved maximum accuracy of 99.1% using random forest model which is superior to the existing mechanism of detection of DDoS attacks.

INDEX TERMS Blockchain, DDoS attack, Federated Learning, Flower framework, Machine Learning.

I.INTRODUCTION

A new technology that has gained popularity globally in recent years is Blockchain due to its unique properties. Some unique fundamental features of blockchain are Security, Immutability, Decentralization, Distributed Consensus, DApps, Smart Contracts, Anonymity, and Transparency. A transparent and immutable platform is created for conducting transactions and storing valuable data in the blockchain. Each block in a blockchain contains a list of transactions, and a blockchain is made up of blocks. The blocks are linked chronologically using cryptographic hashes with the previous block that creates an immutable, tamper-proof record.

Since blockchain technology is decentralized, there is no central authority. All the data are encrypted in nature, making it resistant to attacks. However, some potential deficiencies and attack techniques are still used. There are many attacks on the blockchain, like a 51% attack, Finney attack, Eclipse attack, Block withholding attack, Sybil attack, Time jacking attack, DDoS attack, Selfish mining attack, Race attack, etc. A 51% attack occurs

when one entity or a group of conspiring entities possesses more than 50% of the network's mining power in the network that deploys the Proof of Work (PoW) algorithm. By controlling the transaction validation process, the attacker can double-spend or block specific transactions from being included in a blockchain. A Sybil Attack is an attack that involves setting up several fake identities or nodes to take control of the blockchain network. It influences the network's consensus procedure. Eclipse Attack involves surrounding a target node with rogue nodes, which takes it under the attacker's control to isolate them, make them receive inaccurate information, and prevent them from participating in the consensus procedure. Blockchain networks are susceptible to DDoS attacks where attackers can stop the blockchain from operating normally, processing transactions sluggishly, or even temporarily shutting down the network by flooding it with lots of traffic.

Traditional attack detection methods are ineffective because of the ever-increasing size of the blockchain and the deployment of new contracts on a blockchain network. Many recent attacks are being generated on a day-to-day basis. The use of machine learning algorithms for attack detection can be used to detect the attack effectively. Machine learning [1] can analyze vast volumes of data and identify patterns, irregularities, and potential risks in real time. By examining the data flow, network packets, and communication patterns, these algorithms can analyze suspicious activities or trends that differ from the regular packets. Using these techniques, numerous persistent attacks on blockchain like DDoS, DoS and 51% attacks can be identified. Machine learning algorithms can also be used for anomaly detection, real-time monitoring, pattern recognition and behavioral analysis.

However, the different patterns of attack data collection from various places and training the model are practically tricky. This challenge is solved by using distributed federated machine learning. Under federated machine learning, distributed nodes take local attack data and create a local model by training [2]. Thus, the local models are aggregated at a central server to make a more robust global model that will be used for attack detection. Employing a more distributive node for training is costly, which is also solved in the proposed system by integrating the attack detection in the blockchain network and employing blockchain miners to train the local model. This article focuses on DDoS attack detection using federated machine learning.

The proposed model has the uniqueness of utilizing the already available resources of miners to detect the DDoS attack, which is claimed as integrating a federated machine learning mechanism [3] for detecting DDoS attacks in the blockchain. Federated machine learning is added in the data layer of the blockchain in the place of the block creation and request handling module. After developing a machine learning model to detect the attack with the help of miners in the blockchain to protect the developed global model from any security attack, the model is stored in the blockchain to secure the machine learning model from any tampering or alterations of the model as it might lead to failure of the model. This type of integration supports each other to detect DDoS Attacks, which is the uniqueness of the proposed integrated method.

The contribution of the article is as follows.

- The proposed mechanism uses federated learning to protect the blockchain attack. There is a possibility of federated machine learning model to be attacked by the attacker. This scenario is eliminated by making blockchain to protect federated learning. So, the reliability of DDoS attack detection is more.
- To improve security and reliability the miners are utilized for federated machine learning model training, block mining, and attack detection.
- The proposed federated machine learning model is used to protect attack against blockchain which may be attacked. There is a possibility of the mining node becomes attacker .To solve this issue a miner selection mechanism based on reputation value is proposed to select the optimal set of miners which trains the attack detection model to ensure more secured model development (avoid malicious users in model development). This enhances security in the blockchain , saves energy and computational power.
- To balance between exploration and exploitation in utilizing data for training the model, the reputation-based incentive mechanism of blockchain is proposed to provide incentives to all miner nodes, for training the local model which enable new attack data to be exposed to the model and make more reliability.
- The proposed miner selection and incentive calculation method are designed to balance exploitation and exploration to avoid a 51% attack and to bring diversity in data training.

The remaining part of the article is organized as follows: Section II presents related work, Section III presents the proposed framework for DDoS attack detection using federated machine learning; Section IV presents the Dataset; Section V discusses the Experimental setup and Performance Measure, VI concludes the article with a conclusion, and VII presents the future work.

Table 1 provides the list of abbreviations used in the paper.

TABLE 1. Abbreviations.

Abbreviations	Definition
DDoS	Distributed Denial-of-Service
DoS	Denial-of-Service
ML	Machine Learning
DL	Deep Learning
FL	Federated Learning
SDN	Software Defined Networking

SVM	Support Vector Machine
MLP	Multilayer Perceptron
CIC	Canadian Institute for Cyber security
Relu	Rectified Linear Unit
SCADA	Supervisory Control and Data Acquisition
TP	True Positives
FP	False Positives
PoW	Proof Of Work
IoT	Internet of Things
IIoT	Industrial Internet of Things
PUF	Physically-Unclonable-Functions
D2D	Device-to-Device
OBU	On-Board-Unit
FML	Federated Machine Learning
DRL	Deep Reinforcement Learning
DNS	Domain Name System
SQL	Structured Query Language
UDP	User Datagram Protocol
LDAP	Lightweight Directory Access protocol
SNMP	Simple Network Management Protocol
NTP	Network Time Protocol
SSDP	Simple Service Discovery Protocol
TFTP	Trivial File Transfer Protocol

II. Related Work

A few attack detection mechanisms based on IoT, SDN and Blockchain are summarized in the literature below.

A. Detecting DDoS Attack in IoT

The integration of blockchain and IoT enabled significant gain for society. A Distributed Denial of Service (DDoS) attack on an IoT blockchain network was performed on a vulnerable mining pool. The study [4,5] suggested performing an Intrusion Detection System (IDS) on fog computing to identify the attack. Training fog nodes allows performance evaluation based on Random Forest (RF) and XGBoost (Improved Gradient Tree Boosting System). The results show that Random Forest can detect multi-attacks while XGBoost can detect only binary attacks.

Edge-enabled IoT systems may utilize distributed blockchain-based security architecture to improve their defence against hackers in today's industries [6]. To determine the success of an attack on an edge-based network, a probabilistic model is created that considers network-level attacks (IoT, Edge), hardware-level attacks, software-level attacks (wallet, smart contract), and blockchain network-level attacks. This work examines sixteen different cyber-attacks included in the suggested attack model. According to their findings, blockchain-enabled edge networks are susceptible to malicious attacks due to the following factors: (i) attack location (ii) attack type (iii) consensus algorithm.

This study introduces [7] the arbiter PUF paradigm in a permission-based blockchain that protects the critical pairs of IoT devices. A new ensemble technique based on machine learning offers a low false-positive and a high detection rate to create a collaborative detection system to identify DDoS attacks on IoT devices.

Devices used in smart homes are susceptible to several threats. The paper presents [8] a lightweight authentication technique that permits secure D2D interactions in a smart home by leveraging the possibilities of the Ethereum blockchain and smart contracts. The work also employs an authentication method and a single server queuing system model to prevent DDoS attacks by limiting the volume of service requests processed by the system.

Distributed Denial-of-Service (DDoS) attacks are launched against IIoT devices, with fatal consequences. A multi-point cooperative DDoS defence technique is proposed for IIoT [9]. A cooperative DDoS defence paradigm for the real-world multi-point scenario using the blockchain to distribute defence data throughout the network safely is presented.

B. Detecting DoS Attack in SDN

The spread of new technologies in recent years, like Software Defined Networking (SDN), has increased the plethora of platforms for Distributed Denial of Service (DDoS) attack generation and it created new potential sources for more advanced DDoS attacks on the intended targets. A study suggests [10] and classifies cutting-edge blockchain-based DDoS mitigation systems. The DDoS mitigation strategies are divided into four categories: near-attacker, network-based, near-victim, and hybrid solutions in the network architecture for SDN and IoT. The study's findings explore the difficulties in conducting research and the potential paths for implementing blockchain-based DDoS mitigation solutions.

A thorough analysis of the DDoS detection and mitigation techniques currently used in SDN is reported [11]. Before proposing potential areas for future research, the work also covers current issues with SDN security and the applications of various security solutions.

Supervisory Control and Data Acquisition (SCADA) systems are used to supervise and monitor critical infrastructure and industrial operations [12]. Combining current SCADA systems with SDN has resulted in the creation of numerous SCADA systems based on SDN. A SCADA system faces various attacks because of their wrong deployments. SDN-based SCADA systems using Recurrent Neural Network (RNN), which includes Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU), are recommended for detecting DDoS attacks in the study. The

experimental work produced a DDoS attack detection accuracy of 97.62.

DDoS attacks can be detected using CMCNN [13] to improve generalizability and to lower the model's incidence of reporting errors. Abstract properties of attack traffic are extracted using a stacked sparse auto encoder based on the Kullback-Leibler divergence parameter encoding. The model's sparsity is altered by reducing the redundant data and coupling the abstract features. Encoded feature outputs are then successfully classified. An improved random gradient-descent technique is used for global parameter optimization, preventing fluctuation of the training parameters and speeding the convergence of the model.

Cochain-SC is used to mitigate intra-domain and inter-domain DDoS attacks utilizing SDN [14], blockchain, and smart contracts. It can efficiently reduce DDoS attacks within the target's domain while sharing attack information, making it an effective solution across multiple domains to mitigate DDoS attacks and to combine an intra-domain approach based on SDN with an inter-domain scheme. Cochain-SC has 100% detection rate at 500 Mbps and a False Positive Rate of 23%.

Co-IoT is a collaborative DDoS attack mitigation technique employing smart contracts to promote attack collaboration among SDN-based domains [15] and to transfer attack data securely and efficiently. The attackers try to gain control over many compromised IoT devices and launch DDoS attacks against the victim. Co-IoT enables efficient DDoS mitigation along the path of current attacks and also provides mitigation near the source of the DDoS attack.

ChainSecure is used to protect blockchain nodes from DNS amplification attacks, which are a sort of DDoS attack. It is a scalable and proactive solution in Software Defined Networks. It comprises of three schemes [16], namely StateMap, which performs a one-to-one mapping between DNS request and response. Then, an Entropy calculation scheme that uses sFlow that is used to measure the disorder in data which can detect illegitimate flows and DNS DDoS Mitigation that effectively mitigates unlawful DNS requests. The results of the experiments reveal that ChainSecure protects blockchain nodes and can detect/mitigate the attacks fast, with high accuracy making it a promising solution to protect blockchain nodes against DNS amplification attacks.

Designing the blockchain layers is challenging, especially since the consensus layer is crucial. Identifying Long-Range Attacks (LRA) on Proof-of-Stake (PoS) is a tedious task. Previous research has revealed difficulties in spotting far-reaching attacks and keeping track of validator node operations on the blockchain network. To prevent long-range attacks, a study [8] suggested the classification of nodes on Proof-of-Stake using Deep Learning algorithms.

C. Detecting DoS Attack in blockchain.

Brainchain employs four strategies to safeguard the permissioned blockchain from DDoS attacks [18]. Because blockchain has restricted nodes, attackers can easily target it. They use a real-time detection system called Bayes Network-based Filtering scheme (BF), which can detect anomalies in the network automatically. This method protects blockchain nodes without altering the existing software or the blockchain nodes. Because of its real-time detection scheme, it can detect 100% attacks with less false positive rate.

A new method is proposed to increase security and privacy [19] during exchange of threat intelligence using Blockchain and Federated Learning to share threat detection models as cyber threat intelligence. Threat detection models use federated learning technologies for scalable machine learning applications. Users can access a trained threat detection model without disclosing personal information to the central server. Blockchain technology logs OBU's actions to stop some attacks from tampering with the data [4]. Deep Reinforcement Learning (DRL) technique is suggested to minimize the incentives for activating OBU learning in a dynamic environment and making intelligent judgments.

Sometimes, sharing the training data leads to privacy issues as users may not be interested in sharing their data, and this also leads to a centralization problem where a single entity will control the data. Critical information on blockchain transactions or smart contracts can be secured while performing meaningful analysis and identifying attacks or abnormalities using federated learning. Thus, Federated Machine Learning (FML) enables many nodes to jointly train a machine learning model without sharing their raw data. By doing so, privacy issues and data ownership problems can be overcome while utilizing all the nodes accumulated information. The article intended to develop an integrated federated blockchain framework with reputation-based miner selection [20] and incentive mechanism for analysing the following main hypothesis. Will an integrated blockchain and federated learning framework protect from DDoS attacks and mutually help each other? Will reputation-based miner selection and incentive mechanism improve the security of blockchain and whether to use federated machine learning or not?

Thus, the related work elaborates on the previous work done by various researchers.

TABLE 2. Comparison of the existing work.

Research work Author name/citations	Method	Result achieved	Research gap
Zhang and Kumar	Random forest and XGBoost .	XGBoost -detect only binary attacks. RF can detect multi-attacks.	While training machine learning model can attacked which is not addressed. The miner may be attacker which is not focused
Halgamnge	Probabilistic model is created.	Can detect 16 types of attacks.	Attack during training machine learning model not addressed . Miner attack not addressed
Huang	SDN based SCADA system deploys 3 algorithms namely RNN, LSTM, GRU.	DDOS attack with 97%	Attack during training machine learning model not addressed . Miner attack not addressed
Proposed Framework	Random Forest, Multilayer Perceptron, Logistic Regression and SVM.	Able to detect DDOS attack with 99%	a miner selection mechanism based on reputation value prevent attack from miner .Since the selected miner only involved the training of the model ,the attack during training is solved . The blockchain protect attack on trained machine learning model . The posed model is more secure and reliable

III. Proposed framework

Due to its secure, immutable, anonymous, and data integrity feature, blockchain has been considered a ground-breaking and revolutionary technology. A DoS (Denial of Service) attack in a computer network or ethical hacking could take the following forms: requests overloading the ports, takeover of web servers, refusal of all forms of authentication, and refusal to accept any online services. A DDoS attack is a type of distributive DOS attack. There are two methods for conducting DDoS attacks:

1. Transaction flooding DDoS attacks: Transaction flooding is one of the primary DDoS attacks in blockchain. A hacker may jeopardize the accessibility for authorized (original) users by flooding spam and fake transactions on the network. Malevolent attackers can fill the entire block with fake or spam transactions by sending repeated transactions to the blockchain network, preventing permitted transactions from being added.

2. Smart contract-based DDoS attacks: A DDoS attack can also be directed at a smart contract in several other ways by sending the intelligent contract a computationally costly transaction that stops other transactions from being included in the current block. Another technique is to develop a contract that consumes all the gasoline automatically and makes the service unavailable to other users. Federated machine learning is used to detect such DDoS attacks.

There are many deep learning-based approaches presented in the literature for detecting DDoS Attacks, but the proposed method has significant differences compared with the existing deep learning models in the following aspects.

1. The existing deep learning models are centralized training models that can only learn a few attack pattern data which is collected from a single point of source which makes the model less reliable and not robust.But the proposed framework applies distributed learning called federated machine learning and the attack data are captured at multiple points and multiple places leading to large volume of data which makes the model more accurate and robust.
2. The existing deep learning models developed for DDoS attacks do not have any security measures or mechanisms to protect the machine learning model itself from attacks like poisoning attacks. These attacks may reduce the accuracy or fail to do their intended job of detecting DDoS attacks. In the proposed framework, the security measures are taken to protect the deep learning model from attack.
    i. To avoid attack during the training of the deep learning model, mining nodes are used for training, which is selected with the proposed reputation based miner selection and incentive mechanism. It ensures the training nodes are genuine and the data utilized in the model is attack-free.
    ii. In existing federated machine learning mechanisms, there is a chance of attack while sharing the local model from the client node to the aggregating node. To avoid attacks in the federated machine learning model while sharing the local model from the client to the aggregating node, the models are shared through immutable blockchain transactions.
    iii. After the training process is over inorder to protect the trained model from attack, it is stored in the immutable blockchain storage.

Federated machine learning is selected because of the following advantages.

1. Data Privacy: FML eliminates the requirement for participants to exchange raw data with a centralized authority and allows them to keep their sensitive data local. By exchanging only model updates, privacy issues are decreased.
2. Data Ownership: Participants maintain ownership and control over their data, which can be crucial to companies subject to regulations or when customer data privacy is vital.
3. Lower Communication Costs: Communication costs are much lower because only model changes are transmitted. This makes it useful for devices with limited resources or networks with constrained bandwidth to participate in the network.
4. Distributed Computation: FML uses the combined computing capacity of all participants, allowing for practical parallel training and quicker convergence of the global model.

In this work, the following stages are included in a typical federated learning setup:

1. Model Initialization: A central server initializes the global model using a pre-trained or specified architecture.
2. Local Model Training: Individual participants train models using local data rather than sending data to a centralized server. The participant-specific datasets are used to train the local models, protecting data privacy.
3. Model Aggregation: The participants provide locally learned models or model updates to the central server. The server combines various models to produce an updated global model using safe multi-party computation or federated averaging methods.
4. Iterative Training: Steps 2 and 3 are performed several times to allow the global model to be developed.

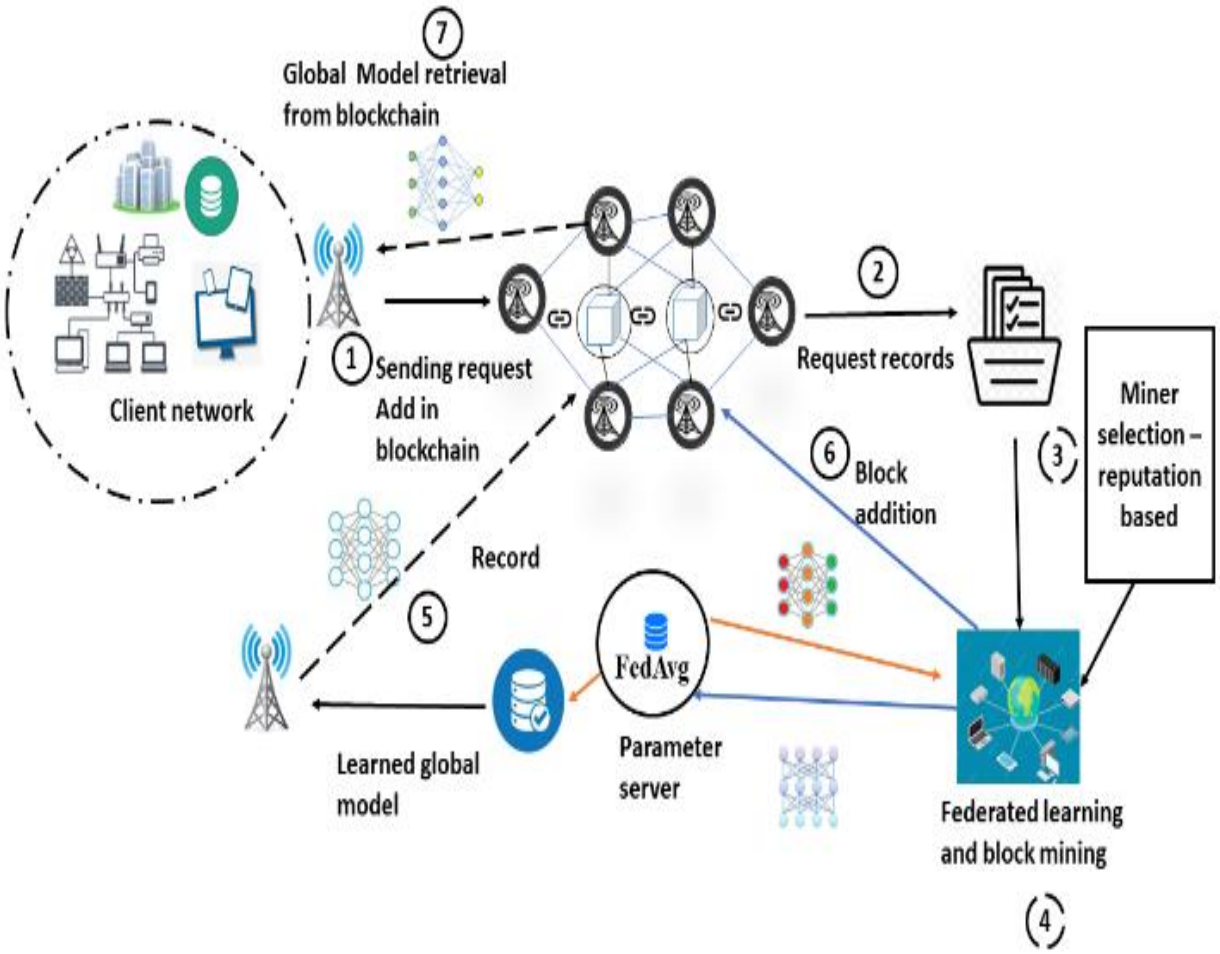

FIGURE 1. Proposed Federated Machine Learning-Based DDoS attack detection framework

Figure 1 shows the proposed blockchain DDoS attack detection framework using federated machine learning.

It is evident that in the already existing blockchain model few changes have been incorporated to detect DDoS attack. As a first change, the miners are involved for both mining the

blockchain and as well as to train the machine learning model to detect the DDoS attack. All the miners act as a cooperating node for developing a federated global machine learning model. The developed ML model is secured from the attack by storing them in the blockchain.

The proposed work consists of the following components.
1. Federated Machine Learning for DDoS detection
2. Blockchain
3. Optimal Miner Selection
4. Reliable DDoS detection

The significance of each component is as follows.

1. Federated Machine Learning is one type of collaborative distributed machine learning which will be more suitable for detecting distributed denial of service attacks on the transaction to the blockchain. But one of the drawbacks of federated machine learning is extra communication overhead from each client to obtain the model from the server.

2. Blockchain technology which enables immutability of the transaction and security of the transaction. Here in the proposed work blockchain is used to store the developed federated ML model securely in addition to storing the client transaction data.

3. Optimal Miner Selection: In traditional blockchain mechanism many nodes or all the nodes in the network will try to act as a miner to make blockchain transactions by solving the cryptographic puzzle. In this approach every node participates in solving the puzzle, but which miner solves the puzzle first is declared as the winning node and allowed to mine the blockchain transaction. In the traditional approach, all the nodes are wasting computation power and energy by trying to solve the puzzle. This wastage of

computing power and energy is avoided by means of optimal miner selection by the proposed reputation-based mechanism.

4. Reliable DDoS detection: Reliability of the detection is realized by the proposed framework by integrating blockchain and federated machine learning and involving miner as trainer for developing the model. Here blockchain is used to store the developed ML model in a secure manner were attacking of the model is avoided, and the reliability of detection through the model is ensured. The reputation-based miner selection ensures only legitimate miners are involved in the ML model development which in turn ensures the reliability of the developed ML model.

It uses blockchain miners for Federated machine learning, i.e., local model training. This miner usage for Federated machine learning prevents attackers from training the local model. As the second layer of secure training, the miner is also selected based on a reputation value, which is updated dynamically. The first novelty is the reputation that is calculated based on quality data. It can be observed from Algorithm 1 under model aggregation update that the reputation function consists of 3 parameters (acc, stake, no. winning time). Acc is the accuracy generated by the individual node during the training. The second parameter is stake (i.e.) how much stake it has in the network (PoS), and the last parameter is the number of winning times (the number of times the miner is a winner) in the mining process of blockchain, which indirectly indicate the loyalty and correct training ability of the individual node. Any miner producing high accuracy, more stake value, and many times it has won will get more reputation value and will be selected as the optimal miner. If a miner only has correct and quality data, it can win more times and produce a more accurate value.

Reputation updating is periodically carried out whenever a new block generation request is created in the blockchain environment. In the algorithm, it can be observed that reputation updating is carried out after one round of training is over when the aggregation of the training model is done at the server. (i.e.) every t, whenever the server model is upgraded to generate a global model, the reputation value for every node in the blockchain scenario is updated.

The second novelty part of the proposed training mechanism balances exploitation and exploration in the training process and avoids 51% attack.

The existing mechanisms do not provide incentives to all the participants of the blockchain. However, the proposed work provides every participant based on reputation value incentives. This enables more participants to participate and be involved in more quality data training. Otherwise, only a few selected miners will be exploited in the blockchain. Since everyone gets an incentive, it balances exploitation and exploration in the optimization process. The proposed framework of reputation-based incentives for all also ensures that there will not be any monopoly in mining and training the local model. It can avoid 51% attacks in the blockchain. This also avoids attacks in the local model training of federated machine learning.

The system flow is described as follows.

1. The clients want to create blockchain blocks and will send the data for block creation in the blockchain.

2. The request is stored in the request queue as a request record for mining purposes.

3. The proposed miner selection procedure based on reputation is executed, and a set of optimal miners is selected for the block.

   The reputation score of miner M at t+1 instant is

   $$\mathbf{r_M(t+1) = r_M(t) + (A_M(t+1) - A_M(t)) + (S_M(t+1) - S_M(t)) + fWt_M(t)/(t - t_0)} \quad (1)$$

   The optimally selected miners start to mine the block based on the stake, solve the puzzle, and train the local model using local attack data.

4. All miners do local training for the local model and transfer it to the parameter server to average the local model. The average local model is created as a global model. The global model is transmitted to the blockchain network to create a block that consists of the global model. This ensures and avoids the global model from attack. This global model block generation request is stored in the record and mined with the selected miner. Finally, it is stored as one of the blocks in the blockchain.

5. Update of reputation and incentive allocation: All participating nodes create the local model and are provided with an incentive.

   The reputation value is updated using the following formula.

   Incentive allocation:

   $$\mathbf{I_M(t+1) = \alpha\,(A_M(t+1) - A_M(t)) + \beta r_M(t)} \quad (2)$$

   Reputation update:

   $$\mathbf{r_M(t+1) = r_M(t) + (A_M(t+1) - A_M(t)) + (S_M(t+1) - S_M(t)) + Wt_M(t)/(t - t_0)} \quad (3)$$

   6. The winning miner will mine the block and create the block that is at the top of the request queue.

   7. Once the global model is stored in the blockchain. All client nodes and miners will read

out the global model and utilize it to detect the DDoS attack on their premises.

The proposed framework ensures there is no possibility of attacking the local model generation and the global model. This secured global model also protects the blockchain from DDoS attacks.

Algorithm 1. Federated Learning for DDOS Attack Detection (Provides the algorithm of the proposed framework as pseudo code.)

1. Input: initial local m M_{lo} (pre-trained model), Batch size S_B, No of batched N_b, Number of selected Miners N_M, M-list of miners, Number of epoch N_{ep},
2. Output: G_mGlobal model for DDOS Attack Detection
3. **Initializing: (t=0)**
4. Select N_M number of miners list L_M based on reputation score.
5. $L_M = \max_M(repute(acc, stake, no.winning\ time))$
6. Update the global weight matrix W in the Parameter server update (W)
7. Define initial values = S_B, N_{ep}, K
8. Initial local models (M_{lo} (wl))
9. Model (M_{lo} (wl)) = set parameters $(wl_1, wl_2 wl_3 .. wl_n)$ for l=1,… N_M
10. **Client model update:**
11. Local models, cryptographic puzzle → $selected\ miners\ L_M$(PoS)
12. **Federated Training at each miner:** // beginning FL method.
13. Receive from parameter server (local models M_{lo} (wl))
14. For local epoch N_{ep}
15. For each batch, 1 to N_b
16. Run local models M_{lo} (wl) on attack pattern data.
17. Run mining process. (PoS)
18. Obtain and set model parameters $wl_1, wl_2 wl_3 .. wl_n$ and cryptographic puzzle solution C_{FA} timestamp T_{ml}.
19. Return (Model parameters & puzzle solution (timestamp).
20. Parameter server = $edAvg\_rep\_up$ (W_{il}, C_{FA}, T_{ml})
21. $\boldsymbol{FedAvg}$: // model aggregation
22. Server (Initialize W)
23. Update repute (acc, stake, no. winning time) for all miners L_M
24. Update global model $G_m(W) = AVE(W_{il})$
25. Send global model $G_m(W)$ to all client Miners.
26. Evaluate and find the winner of miners.
27. Send approval for the winning miner to generate a block.

IV. Dataset

The dataset used for the research is taken from Kaggle. Two datasets are used in the study namely IDS 2018 Intrusion CSVs (CSE-CIC-IDS2018) and (CIC-DDoS2019). The University of New Brunswick originally developed this dataset to study DDoS attacks. The dataset is based on server logs of the university, which revealed several DoS attacks during the period made available to the public. The dataset comprises benign and the latest common DDoS attack data, mirroring real-world pcap. Additionally, it provides network traffic analysis outcomes by utilizing CICFlowMeter-V3. Reflective DDoS attacks like SYN, UDP, DNS, PortMap, NetBIOS, LDAP, SNMP, UDP-Lag, NTP, and MSSQL were employed. These attacks were subsequently carried out over two days. A total of 12 DDoS attacks were executed on the training day like SYN, UDP, DNS, NetBIOS, LDAP, SNMP, UDP-Lag, NTP MSSQL, SSDP, Web DDoS, and TFTP. On the testing day, seven attacks were conducted, including MSSQL, UDP, UDP-Lag, Port Scan, NetBIOS, LDAP, and SYN. This dataset consists of eighty columns, each corresponding to a record in the IDS logging system used by the University of New Brunswick. Destination port, protocol, flow duration, total forward packets, total backward packets, and label are the most significant columns that are used in the study.

Two approaches were used for implementing federated machine learning. The server-client framework from scratch is developed using TensorFlow federated programming high-level APIs, and the second approach uses flower framework for federated learning. A federated learning system called Flower is used because it is simple to use and can be expanded. It is written in Python and interoperable with PyTorch, TensorFlow, and Scikit learn, among other machine learning frameworks. The following characteristics of Flower make it a good option for federated learning:

- A straightforward API that makes it simple to begin using federated learning.
- A range of machine learning algorithms are supported.
- The capacity to train models on various hardware, including servers, laptops, and mobile phones.
- Support for secure aggregation, which aids in preserving user data privacy.

V. Experimental Setup and Performance Measure

The proposed federated machine learning uses five client nodes to implement the system. Three machine learning algorithms are deployed to test the performance of the proposed system, namely Random Forest Classifier, Multi-Layer Perceptron Classifier, and Logistic Regression. The

parameters used for the experimental study are presented in Table 2. The parameters are fixed experimentally on a trial-and-error basis for achieving highest accuracy. All the parameter values from minimum to maximum values are tried and the value which achieves maximum accuracy is chosen as the final parameter.

TABLE 3. Experimental setup.

Model	Parameter
Federated framework: Flower	Number of nodes=5
RF	1.number of trees=100, 2. criterion="Gini"; 3.min_samples_split=2; 4.max_depth=till convergence. 5.min_samples_split= sqrt(n_features)
MLP	solver='lbfgs', alpha=1e-5, hidden_layer_sizes=(5, 2),
LR	Tolerance for stopping criteria= 1e-4, max_iterint=100, solver='lbfgs'
Aggregation Method	Average
Consensus Protocol	Modified Proof of Stake
Reputation update instant	At new block request and every iteration during training
Batch size sd	64
No of batches	50
No of epochs	100

A. Theoretical Time Complexity Analysis

Proof of Stake (PoS) consensus mechanism has a time complexity of O (log n). Here, in the proposed algorithm, the research uses a similar kind of Proof of Stake where the reputation value and accuracy of classification value are taken as stake value.

The time complexity of federated learning depends on several factors, including the number of participating devices, the number of local data samples, the model complexity, and the communication overhead.

Local training: The time complexity is typically O(n^2), where n is the number of parameters in the model.

Model Aggregation: The time complexity of aggregating model updates from multiple devices depends on the number of participating devices and the size of the model updates. The aggregation process typically involves collecting updates from all devices, averaging them, and broadcasting the aggregated update to all devices. The time complexity of Federated Learning is typically O(pN), where,p is the number of parameters in the model, and N is the number of participating devices. The comparison of the time complexity of Random Forest, Multilayer Perceptron and Logistic Regression is provided in Table 3 as follows:

TABLE 4. Time Complexity of the various models.

Model	Training Time Complexity	Prediction Time Complexity
Random Forest	O(n * log(n) * d * k*p*N)	O (d * k*p)
Multilayer Perceptron	O (n * d * t*p*N)	O (d * t*p)
Logistic Regression	O (n * d*p*N)	O(d*p)

Where

n is the number of training samples
d is the number of features
k is the number of features considered at each split in a decision tree
t is the number of iterations of the stochastic gradient descent algorithm

B. Theoretical Communication Overhead Analysis

The communication overhead in Federated Learning is primarily influenced by the following factors: Model Size, Number of Participating Devices, and Communication Frequency.

Larger models require larger updates to be transmitted, resulting in higher communication overhead. Here, the work updates p parameters as local model update to the aggregating server since n is the number of devices it is communicating with, and the communicating overhead is O(p*N). Since the model is trained using t iteration, it is a number of communications is involved. Two communication overheads are involved to communicate the initial and final model to the clients. So, the overall communication overhead is O(P*N*t+2) packets. The average accuracy acquired at the parameter server for various machine-learning algorithms is shown in Table 4 for centralized and decentralized system.

TABLE 5. Performance Measurement.

Model		Accuracy (in %)	Recall (in %)	F1_Score (in %)
Random Forest Classifier	Centralized	98.7	99.1	99.3
	Decentralized	99.1	99.3	99.5
Multilayer Perceptron Classifier	Centralized	94.5	95.1	95.4
	Decentralized	95.1	95.2	95.6
Logistic Regression	Centralized	88.0	87.4	88.7
	Decentralized	88.5	87.6	88.9
SVM	Centralized	87.9	87.1	88.5
	Decentralized	88.0	87.5	88.8

Table 4, shows Random Forest classifier performed well, with an accuracy of 99.1% compared to the other models. Multilayer perceptron classifier performed next with an accuracy of 95.1%. The Logistic Regression performed poorly with 88.5% accuracy, as shown in the graph below.

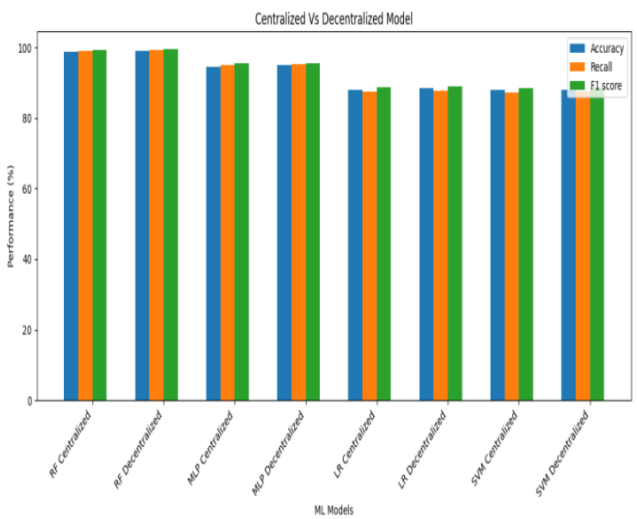

FIGURE 2. Performance Measurement of centralized / decentralized ML models.

Figure 2, gives performance measure of centralized and decentralized machine learning model deployment. The performance is slightly higher in decentralized approaches because of diversified learning taking place in the decentralized learning mechanism.

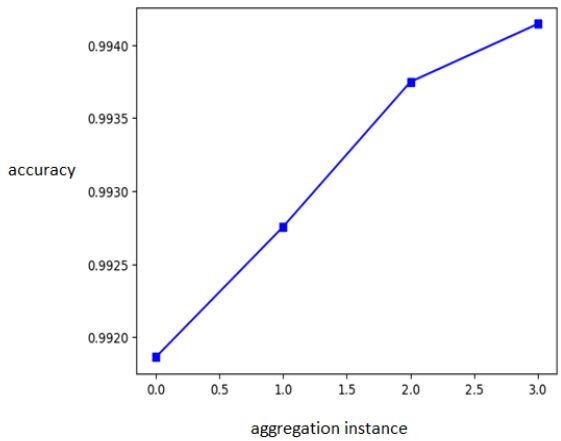

FIGURE 3. Random Forest Classifier aggregation performance

Figure 3, shows the aggregator performance measure at the parameter server where the model performance improved because of aggregation of the locally trained model. It shows the sample of aggregation at four different instances of time. While aggregating, Random Forest provides increased accuracy over the period of aggregation only. This is happening because the Random Forest classifier is one type of ensemble machine learning model.

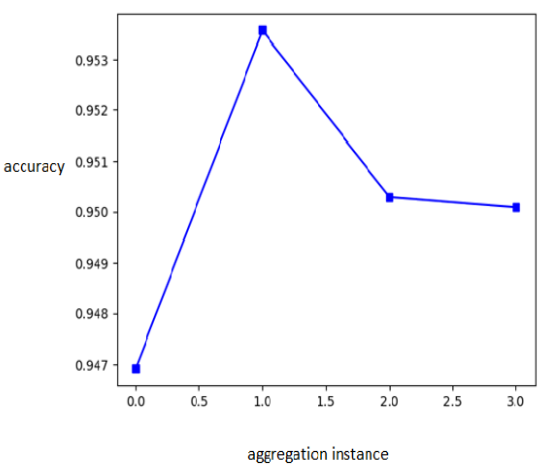

FIGURE 4. Multilayer Perceptron Classifier aggregation performance

Figure 4, shows the multilayer perceptron's performance improvement at the parameter server. After collecting and aggregating a few locally trained models, the model's performance is going up and down. The performance went a slight dip and is settled down to a value. The initial increase of the model performance happened because of the aggregation as the model gets more shared knowledge from

the locally trained models. After a few instance of aggregation the model starts to over fit which makes the model accuracy decrease.

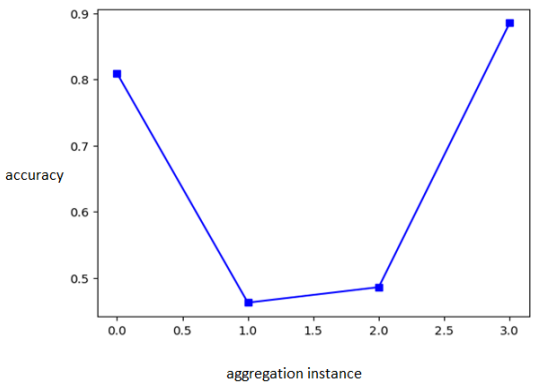

FIGURE 5. Logistic Regression aggregation performance

Figure 5, shows the aggregated global model performance at the parameter server, where the performance decreased slightly while aggregating the locally trained models. It increased to maximum value during the aggregation process. There is a decrease then increase of aggregation performance in logistic regression because initially the Logistic Regression model is under fit then over the process of aggregation the model is improved.

The performance graphs of Figures 3 to 5 show that only the Random Forest classifier's performance keeps improving because of the locally trained model aggregation. The global model's performance went up and down for all other models due to inaccurate model training.

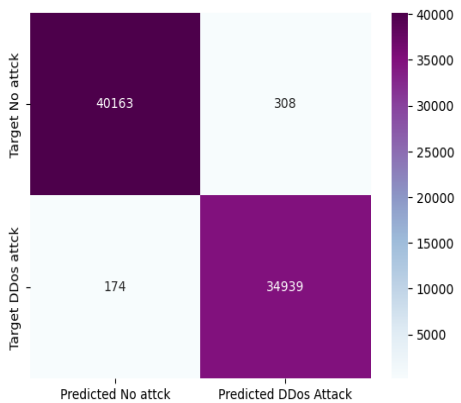

FIGURE 6. Confusion Matrix

Figure 6, shows the confusion matrix obtained for the binary classification of a DDoS attack. The study focuses on the binary classification of DDoS (Distributed Denial of Service) attacks, explicitly differentiating whether a DDoS attack is present or absent. The matrix comprises two rows and two columns, encapsulating the model's predictive capabilities. In the first row, the model correctly identified most non-attack instances, represented as True Negatives (TN) with a count of 40,163. However, it's also essential to note that the False Positives (FP) totaling 308 are found in the same row. These FP instances denote situations where the model mistakenly flagged benign traffic as DDoS attacks, highlighting the need to fine-tune the model to reduce false alarms. In the second row, it is observed that the model occasionally missed DDoS attacks, represented as False Negatives (FN), which occurred 174 times. This suggests room for improvement in sensitivity to detect actual attacks. However, the model did manage to correctly identify DDoS attacks, as indicated by the True Positives (TP) count of 34,939 in the same row. While it's promising that the model exhibited high accuracy in identifying attacks, further optimization may be beneficial to enhance its ability to distinguish malicious attacks from regular traffic.

C. Comparative Analysis for different numbers of samples

The model is compared with CMCNN [12] by examining various data flow samples for regular and DDoS attacks in a simulation environment. The model's tendency to identify DDoS attack traffic was examined for multiple network traffic samples. (For samples like 0, 5000, 10,000, 15000, 20000, 25000, and 30000). The Recall, Accuracy, and F1_score values of the two approaches are shown in Figure 7, Figure 8, and Figure 9, which have an upward trend as the training samples grew. The recall rate of the proposed model was very similar to the existing CMNNN model for small sample sizes. As the number of samples increased, the proposed model outperformed CMCNN with respect to Recall, Accuracy, and F1_Score value. It proves that the proposed model can detect DDoS attacks better than CMCNN.

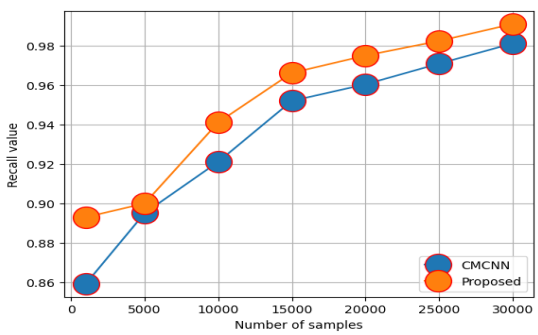

FIGURE 7. Number of samples and Recall value for the proposed and existing model.

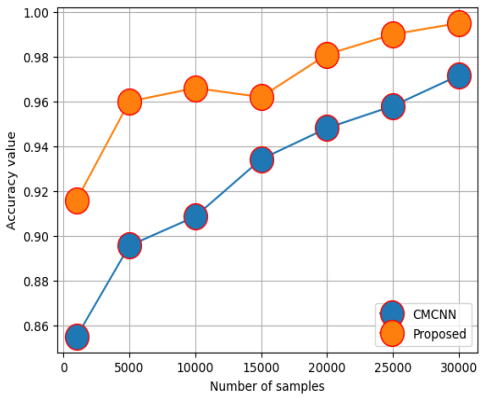

FIGURE 8. Number of samples and Accuracy value for the proposed and existing model

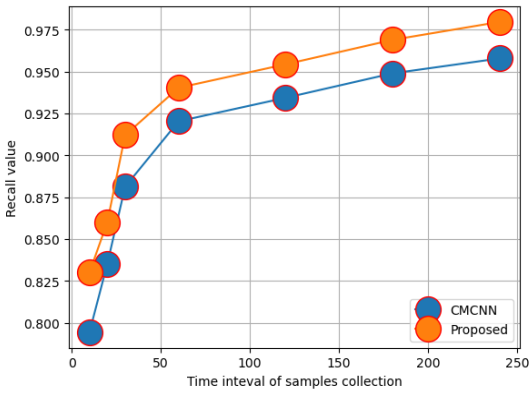

FIGURE 10. The time interval of sample collection and Recall value for the proposed and existing model.

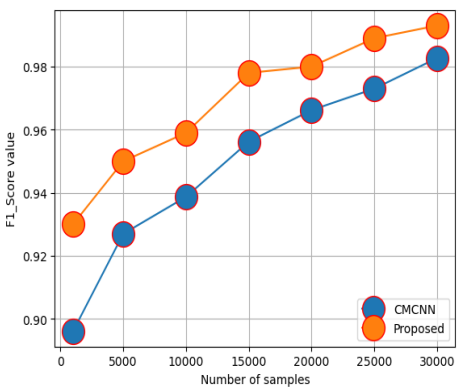

FIGURE 9. Number of samples and F1_Score value for the proposed and existing model

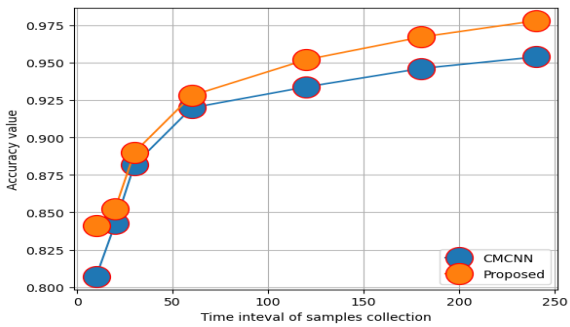

FIGURE 11. The time interval of sample collection and Accuracy value for the proposed and existing model

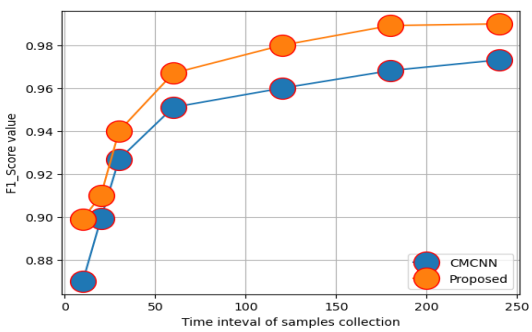

FIGURE 12. The time interval of sample collection and F1_score for the proposed and existing model

D. Comparative Analysis for different intervals of sample collection

The model's detection performance was evaluated while varying the network traffic's time window while maintaining a constant ratio of attack traffic. The collection windows' intervals were measured in seconds. For instance, samples were taken at intervals of 0, 50, 100, 150, 200, and 250.

Figure 10, Figure 11, and Figure 11 show the suggested models produced results with lower accuracy and recall values when sample sizes were small. The recall and accuracy were examined under various window circumstances. As the time interval for samples grew, the recall rate and accuracy of the two models improved, increasing the effectiveness of the detection.

VI. Conclusion

Detecting DDoS attacks in the blockchain is a challenging task, which can be done with a federated machine learning algorithm. Still, the federated machine learning algorithm is prone to attack while transferring the locally trained model to an aggregating server. So, this article proposed a Blockchain-Integrated Federated Machine Learning Algorithm to secure blockchain from DDoS attacks and a

machine learning model from poisoning attackers. This proposed framework ensures the security of both the blockchain and the machine learning model and provides a robust mechanism to detect DDoS attacks in the blockchain. Three machine learning models are compared to solve simple, medium, and complex problems and it is discovered that Random Forest can detect DDoS attacks in a complex environment with high accuracy. Logistic regression achieved an accuracy of 88.5%, and Multilayer Perceptron had an accuracy of 95.1%. The proposed model using Random Forest achieved 99.1% accuracy in detecting DDoS attacks, which is superior to the existing literature. The simulated attack performance of the random forest algorithm is compared with similar literature work. It is observed that for all the performance metrics, the proposed mechanism provides high-performance values. This article addressed only one type of attack called DDoS attack. In future, the same mechanism can be fine-tuned to detect other attacks. Detecting attacks alone will not be a solution for security mechanisms; future mitigation strategies will be developed in a typical practical SDN network environment.

VII. Future Work

Detecting DDoS attacks in blockchain using federated machine learning algorithm has achieved significant results. The same methodology can be deployed to detect other types of attacks that are prevalent in blockchain. A global integrated model to detect all types of attacks can be developed in the near future.

References

[1] Harsh Kasyap, Somanath Tripathy", Privacy-preserving and Byzantine-robust Federated Learning Framework using Permissioned Blockchain", Expert Systems with Applications, vol 238, Part E, 122210,Mar 2024.

[2] Shichang Xuan, Mengda Wang, Jingyi Zhang, Wei Wang, Dapeng Man, Wu Yang", An Incentive Mechanism Design for Federated Learning with Multiple Task Publishers by Contract Theory Approach", Information Sciences, 120330,Feb 2024.

[3] Haoran Zhang, Shan Jiang, Shichang Xuan", Decentralized federated learning based on blockchain: concepts, framework, and challenges", Computer Communications, vol 216, pp.140-150, Feb 2024.

[4] Tian Wen a b, Hanqing Zhang a, Han Zhang a, Huixin Wu a, Danxin Wang a, Xiuwen Liu a, Weishan Zhang a, Yuwei Wang b, Shaohua Cao" , RTIFed: A Reputation based Triple-step Incentive mechanism for energy-aware Federated learning over battery-constricted devices", Computer Networks, vol 241, 110192,Mar 2024.

[5] Sanda, O., Pavlidis, M., Seraj, S. and Polatidis, N", Long-range attack detection on permissionless blockchains using Deep Learning", Expert Systems with Applications, 218, p.119606, 2023.

[6] Muhammad Nadeem Ali, Muhammad Imran, Muhammad Salah ud din and Byung-Seo Kim",Low Rate DDoS Detection Using Weighted Federated Learning in SDN Control Plane in IoT Network", Appl. Sci. , 13(3), 1431,2023.

[7] Francisco Lopes de Caldas Filho, Samuel Carlos Meneses Soares, Elder Oroski, Robson de Oliveira Albuquerque, Rafael Zerbini Alves da Mata, Fábio Lúcio Lopes de Mendonça and Rafael Timóteo de Sousa Júnior", Botnet Detection and Mitigation Model for IoT Networks Using Federated Learning", Sensors , 23(14), 6305,2023.

[8] Huang, H., Ye, P., Hu, M. and Wu, J",A multi-point collaborative DDoS defence mechanism for IIoT environment," Digital Communications and Networks", 9,pp.590-601,2023.

[9] Zhang, B., Wang, X., Xie, R., Li, C., Zhang, H. and Jiang, F", A reputation mechanism based on Deep Reinforcement Learning and blockchain to suppress selfish node attack motivation in Vehicular Adhoc networks", Future Generation Computer Systems,139, pp.17-28,2023.

[10] Jiang, T., Shen, G., Guo, C., Cui, Y. and Xie, B", BFLS: Blockchain and Federated Learning for sharing threat detection models as Cyber Threat Intelligence" , Computer Networks, 224, p.109604, 2023.

[11] Vinay Gugueoth, Sunitha Safavat, Sachin Shetty", Security of Internet of Things (IoT) using federated learning and deep learning- Recent advancements, issues and prospects", ICT Express, Vol 9, Issue 5, pp.941-960,October 2023.

[12] Kumar, R., Kumar, P., Tripathi, R., Gupta, G.P., Garg, S. and Hassan, M.M", A distributed intrusion detection system to detect DDoS attacks in blockchain-enabled IoT networks", Journal of Parallel and Distributed Computing, 164, pp.55-68, 2022.

[13] Halgamuge, M.N", Estimation of the success probability of a malicious attacker on the blockchain-based edge network", Computer Networks, 219, p.109402, 2022.

[14] Babu, E.S., Srinivasa Rao, B.K.N., Nayak, S.R., Verma, A., Alqahtani, F., Tolba, A. and Mukherjee, A", Blockchain-based Intrusion Detection System of IoT urban data with device authentication against DDoS attacks", Computers and Electrical Engineering,103, p.108287,2022.

[15] Randhir Kumar, Prabhat Kumar, Rakesh Tripathi, Govind P. Gupta, Sahil Garg, Mohammad Mehedi Hassan", A distributed intrusion detection system to detect DDoS attacks in blockchain-enabled IoT network", Journal of Parallel and Distributed Computing, vol 164, pp. 55-68,2022.

[16] Ziwei Yin, Kun Li and Hongjun Bi ", Trusted Multi-Domain DDoS Detection Based on Federated Learning", Sensors, 22, 7753, 2022.

[17] Chaganti, R., Bhushan, B. and Ravi, V", A survey on Blockchain solutions in DDoS attacks mitigation:

Techniques, open challenges, and future directions", Computer Communications, 197, pp.96-112, 2022.
[18] Polat, H., Türkoğlu, M., Polat, O. and Şengür, A ",A novel approach for accurate detection of the DDoS attacks in SDN-based SCADA systems based on deep recurrent neural networks", Expert Systems with Applications,197, p.116748,2022.
[19] Qian-yi Dai, Bin Zhang, and Shu-qin Dong", A DDoS-Attack Detection Method Oriented to the Blockchain Network Layer", Hindawi, Security and Communication Networks, Article ID 5692820,2022.
[20] Valdovinos, I.A., Pérez-Díaz, J.A., Choo, K.K.R. and Botero, J.F", Emerging DDoS attack detection and mitigation strategies in software-defined networks: Taxonomy, challenges and future directions", Journal of Network and Computer Applications,187, p.103093,2021.
[21] Zakaria Abou El Houda, Abdelhakim Hafid and Lyes Khoukhi ", BrainChain - A Machine learning Approach for protecting Blockchain applications using SDN", IEEE International Conference on Communications, pp. 1-6, 2020.
[22] Iman Sharafaldin, Arash Habibi Lashkari, Saqib Hakak, and Ali A. Ghorbani ", Developing Realistic Distributed Denial of Service (DDoS) Attack Dataset and Taxonomy", IEEE 53rd International Carnahan Conference on Security Technology, Chennai, India, Oct 1-3,2019.
[23] Zakaria Abou El Houda, Abdelhakim Senhaji Hafid and Lyes Khoukhi", Cochain-SC: An Intra- and Inter-Domain Ddos Mitigation Scheme Based on Blockchain Using SDN and Smart Contract" ,IEEE Access,vol 7, pp. 98893-98907,2019.
[24] Zakaria Abou El Houda, Abdelhakim Hafid and Lyes Khoukhi", Co-IoT: A Collaborative DDoS mitigation scheme in IoT environment based on blockchain using SDN", IEEE Global Communications Conference (GLOBECOM), pp.1-6,2019.
[25] Zakaria Abou El Houda, Lyes Khoukhi and Abdelhakim Hafid ", ChainSecure - A Scalable and Proactive Solution for protecting Blockchain applications using SDN", IEEE Global Communications Conference (GLOBECOM), pp.1-6,2018.

D.Saveetha completed her B.Tech in Information Technology from AMSEC and her M.Tech in Information Technology (Networking) from Vellore Institute of Technology. She is pursuing her PhD in Computer Science and Engineering from SRM Institute of Science and Technology. D.Saveetha is working as an Assistant Professor at SRM Institute of Science and Technology, India. Her area of research includes blockchain, where she focuses on the security aspects of blockchain and its practical implementation in real-time. She has published around 35 research papers in blockchain, security and networking. She is a member of various professional bodies.

Dr.G.Maragatham completed her M.E in Computer Science Engineering from Sathyabama University and her PhD in Computer Science Engineering in 2014. Dr.G.Maragatham works as a Professor at SRM Institute of Science and Technology. She has 25 years of teaching experience. Her research areas include Deep Learning and Machine Learning. She is a member of various professional bodies and an expert committee member in multiple journals. She is currently working on SAMSUNG Prism project. She is also a Program Coordinator for AIML specialization and has set up a state-of-the-art robotics lab at SRMIST.

Vijayakumar Ponnusamy (Senior Member, IEEE) received a B.E. (ECE) degree from Madras University in 2000, a Master's degree in Applied Electronic from the College of Engineering, Guindy, in 2006, and a Ph.D. degree in Applied Machine Learning in Wireless Communication (cognitive radio) from SRM IST, Chennai, Tamil Nadu, India, in 2018. He is a Certified "IoT specialist" and "Data scientist". He works as a Professor with the ECE Department, SRM IST. His research interests are machine and deep learning, IoT-based intelligent system design, blockchain technology, and cognitive radio networks. He received the NI India Academic award for Excellence research in 2015.

Nemanja Zdravković completed his M.Sc. in Electrical Engineering and Computer Science, Scientific Field Telecommunications, at the Faculty of Electronic Engineering, University of Niš in 2012 and his Ph.D. studies at the Norwegian University of Science and Technology (NTNU) in Trondheim, Norway, in 2017, as well as at the University of Niš in 2017, for which he has received a dual Ph.D. degree. Dr. Zdravković has been an Assistant Professor at the Faculty of Information Technologies at Belgrade Metropolitan University (BMU) and, since 2020, head of the Blockchain Technology Laboratory at BMU; since April 2023, he has been the Dean of the Faculty of Information Technology at BMU. Besides teaching activities in the field of computer networks, blockchain technology, and computer architecture, he conducts research that includes the application of blockchain technology in healthcare, cooperative and distributed network analysis, RF and optical telecommunication systems analysis, and information theory. Dr. Zdravković is a member of the Institute of Electrical and Electronics Engineers (IEEE) and serves as a reviewer of the flagship conferences in wireless telecommunication systems.

Original Manuscript ID: Access-2024-12120
Original Article Title: "An Integrated Federated Machine Learning and Blockchain Framework with Optimal Miner Selection for Reliable DDOS Attack Detection "

To: IEEE Access Editor
Re: Response to reviewers

Dear Editor,

Thank you for allowing a resubmission of our manuscript, with an opportunity to address the reviewers' comments.
We are uploading (a) our point-by-point response to the comments (below) (response to reviewers, under "Author's Response Files"), (b) an updated manuscript with yellow highlighting indicating changes (as "Highlighted PDF"), and (c) a clean updated manuscript without highlights ("Main Manuscript").

Best regards,
< Vijayakumar Ponnusamy> et al.

Reviewer#1, Concern # 1 (please list here): please give the respond to reviewers.
Author response: Sorry for the inconvenience. But we have attached the response file, may be due to some settings the file is not viewable.
Author action:

Reviewer#2, Concern # 1 (please list here): All comments have been thoroughly addressed, and this paper is ready for publication. However, the authors must condense their contribution; their response file is not viewable.
Author response: Thank You. For your comments. Sorry for the inconvenience. But we have attached the response file, may be due to some settings the file is not viewable.

Author action: We have modified the contribution in page 2.

Reviewer#3, Concern # 1 (please list here): The authors should explicitly mention the significant contributions of the manuscript. The novelty of the paper is not highlighted- needs a small revision.
Author response: We have made revision in the contributions of the manuscript and the novelty of the paper is highlighted.
Author action: We updated the manuscript by modifying the contribution of the manuscript in page no 2.

Reviewer#3, Concern # 2 (please list here): The advantages and limitations of the compared schemes in relationship with similar schemes is not clear ref. to section II, Suggest the authors to provide a comparative analysis of the various data sets preferably in a table to justify the outcome as claimed to be very efficient, where there is a small confusion..
Author response: We have added a comparison table and listed the advantages and limitations of the earlier schemes .
Author action: We updated the manuscript by adding a comparison table in page 4 and page 5.

Reviewer#3, Concern # 3 (please list here): Please revise the structure of the paper, Redraft all the titles and contents appropriately.
Author response: The paper has been redrafted with titles and contents.
Author action: We updated the manuscript by changing all titles and changing contents appropriately.

Reviewer#3, Concern # 4 (please list here): include discussion and future works section to be more concise while focusing on the overall outcome of the research done- while limitations can be included in future works.
Further, as per the article contents there still exists revision in English grammar and word choice as used- use of we at multiple sections to be avoided.
Author response: We have included the future work in the paper .
We have done all the corrections and grammar check .
Author action: We updated the manuscript by adding future work in page 13.

Reviewer#4, Concern # 1 (please list here): The authors need to add a comparison between the last studies and the proposed system.
Author response: A new comparison table is added to study the existing and proposed system.
Author action: We updated the manuscript by adding a comparison table on page 4.

Reviewer#4, Concern # 2 (please list here): The authors need to add more references (less than 25 Ref,)
Author response: We have added a few more references.
Author action: We updated the manuscript by adding the references in page 13.

Reviewer#4, Concern # 3 (please list here): the picture of figures needs to be more accurate and high resolution such as Fig.1.
Author response: The resolution of the image is improved and drawn again.
Author action: We updated the manuscript by adding the image in page 6.

Reviewer#5, Concern # 1 (please list here): The paper is good for publication.
Author response: Thank You

Author action: Nil

Note: *References suggested by reviewers should only be added if it is relevant to the article and makes it more complete. Excessive cases of recommending non-relevant articles should be reported to ieeeaccesseic@ieee.org*