I. J. Education and Management Engineering, **2026, 3, 94-104**
Published Online on June 8, 2026 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijeme.2026.03.07

Quantum Key Distribution-Enabled Federated Learning over Blockchain for Privacy-Preserving AI in Large-Scale IoT Networks

David Shiala Ongoma*
School of ICT, Media, and Engineering, Zetech University, Nairobi 00100, Kenya
E-mail: davongoma@gmail.com
ORCID iD: https://orcid.org/0009-0008-0290-6291
*Corresponding author

Received: March 25, 2026; Revised: April 28, 2026; Accepted: May 19, 2026; Published: June 8, 2026

Abstract: The proliferation of massive IoT networks has created an environment where distributed AI can be achieved. At the same time, it introduces serious privacy and security challenges. Federated learning (FL) allows training local models on IoT devices and aggregating them without sharing data, but still suffers from problems such as gradient inference attack, Byzantine model poisoning attack and the failure in single point of failure centralized aggregation point. In this paper, we propose QFL-BC, a framework combining Quantum Key Distribution (QKD) and a permissioned blockchain to holistically tackle the problem. Using the BB84 protocol with decoy states, QKD generates a One-Time Pad key to encrypt the model update and achieve information-theoretic security with provable security against a quantum attacker. The central aggregator is replaced by the permissioned blockchain with a smart contract, which ensures an immutable audit trail and distributes the orchestration of FL training decent rally, as well as imposes a penalty on malicious participants by automatic reputation score maintenance. The experiments with MNIST and CIFAR-10 on 100 IoT clients under Non-IID conditions show QFL-BC obtains an accuracy of 96.8% against 41.5% for classic FL under 10% poisoning attack (133% relative improvement). We have tested its robustness across adversary percentages of 10%-40% with accuracy above 87.3% and measured scalability up to 500 clients, showing good degradation, communications overhead of 5.84 MB per round, which is only 12.3% higher than the classic FL and analysed latency and energy to evaluate its feasibility on resource-constrained IoT devices.

Index Terms: Quantum Key Distribution, Federated Learning, Blockchain, IoT Security, Privacy-Preserving AI, Post-Quantum Cryptography, Byzantine Fault Tolerance

1. Introduction

Initially an emerging idea, IoT is now a core part of the digital ecosystem: over 22 billion devices are already connected in smart health care systems, smart vehicles, smart grids, smart industry systems and more [1]. With IoT coupled with AI systems, it is possible to process data collected from IoT in real-time; in fact, IoT devices produce data at a zettabyte scale, and this will continue to grow [1,2]. In traditional FL systems, data are aggregated centrally at the cloud to be fed into an AI model, but this causes serious privacy issues and legal issues governed by GDPR, CCPA and more, as well as costly communications overhead [2].

FL allows distributed IoT devices to jointly train an AI model without transmitting their raw data [3]. The issue of centralisation is well overcome by FL, but it has its own attack surfaces like gradient leakage attack that reconstruct private training sample from model updates, Byzantine attack where compromised clients corrupt the global model with poisoned gradients and a single point of failure as a central server where an attack might focus [4]. Moreover IoT devices are not identical: they may range over a vast spectrum of capabilities (compute, bandwidth, storage, power, etc.) which means FL faces specific attack surfaces [4].

Blockchain offers a decentralised, tamper-proof coordination solution that avoids relying on a trusted third-party aggregator [5]. QKD complements it with information-theoretic security (rather than reliance on computational security, which can be attacked by quantum algorithms) on secure key exchange [6]. No paper to date has combined these two technologies into a single framework to provide a system that supports large-scale FL over IoT.

This paper presents QFL-BC, and we summarise our contributions:

i. Hybrid Confidentiality: We generate one-time pad keys by QKD with the BB84 protocol, including decoy states, and this achieves provably secure model updates encryption to classical and quantum-susceptible eavesdroppers.
ii. Blockchain-orchestrated federated aggregation: By use of a permissioned blockchain with smart contracts to discard the server for data aggregation, we achieve immutable logging of system operations, decentralised FedAvg aggregation, automatically reward delivery, and Byzantine attack defence based on client reputation.
iii. Thorough Quantitative study: We assess the impact of the number of adversaries from 0 to 40%, the scalability up to 500 clients, communication overhead, latency per round, and device power consumption.
iv. Convergence analysis with non-iid data: Under non-iid heterogeneous data distribution on IoT clients by means of Dirichlet partition, FedAvg convergence is studied against the prior work of Q-secure FL.

We organise the remaining of this paper as follows: Section II is on related works and highlights the gap; Section III provides the system and threat model; Section IV designs QFL-BC; Section V illustrates experimental results; Section VI discusses limits and future problems; Section VII is the conclusion.

2. Related Work

Since the confluence of FL, blockchain, and quantum cryptography (QC), a number of relevant research results have been achieved. This section briefly reviews literature in the proximity of QFL-BC and points out the research gap.

2.1. Security of Federated Learning

Existing privacy-preserving FL has been studied extensively. Differential Privacy (DP) adds bounded random noise to gradients to protect local updates, but loses significant accuracy, especially in non-IID settings encountered in IoT deployments [7]. Homomorphic Encryption (HE) adds privacy to FL by encrypting model updates, but carries computational overhead many orders of magnitude higher than plaintext FL [8] for an IoT-compatible framework. Secure Multi-Party Computation (SMPC) distributes the calculation across many devices, but uses many rounds of communication. None of these methods prevents quantum-capable attackers from eavesdropping on the transmission channel. As a recent survey states, post-quantum secure IoT FL is a "still-open research question" [17].

2.2. Blockchain for Federated Learning

In BC-FL, a central aggregator is not needed. Kim et al. Propose BlockFL, where miner nodes perform validation and aggregation of local model updates, and the local updates are permanently recorded on the chain [9]. Lu et al. Further develop on this by applying a smart contract-based incentive for FL in industrial IoT [10]. Sharma et al. Conduct an in-depth survey on blockchain-FL systems in medical care, transportation and smart grid [15]. In all of the above systems, PKI is utilised for channel security, which is not secure against quantum computers of cryptographically relevant capability. Moreover, the impact of BC consensus latency on the energy budget for IoT devices is still not well understood.

2.3. Quantum Key Distribution for IoT

The optimisation of QKD for limited IoT devices has been a rich area of research. Kumar et al. Demonstrated that decoy state QKD protocol can be implemented in the same hardware as that of the power-efficient IoT nodes [11]. Khan et al. Developed Quantum FL that implemented QKD secure client-server communication in FL; however, it used a centralised aggregator and hence did not consider decentralisation and ledger immutability [12]. Very recently, Amiri et al. Have also shown that long-distance QKD can be achieved for heterogeneous IoT at a city scale [21]. To the best of our knowledge, there has been no framework integrating QKD and blockchain-based FL to provide the information-theoretic communication security along with decentralised aggregation.

2.4. Research Gap and Motivation

Table 1 details the functionalities of current frameworks. However, no previous approach integrates: (1) the information-theoretic secure communication mechanism (QKD), (2) the decentralised and tamper-proof aggregation scheme (smart contracts on the blockchain), (3) robustness analysis with a dynamic percentage of adversaries, and (4) the empirical study of scalability and energy consumption for IoT. QFL-BC supports all four capabilities, verified by the experiments shown in section V.

Table 1. Comparison of related frameworks.

Feature	Classical FL [3]	DP-FL [7]	BlockFL [9]	QKD-FL [12]	Proposed QFL-BC
Decentralized	No	No	Yes	No	Yes
Quantum-Safe Channel	No	No	No	Yes	Yes

Feature	Classical FL [3]	DP-FL [7]	BlockFL [9]	QKD-FL [12]	Proposed QFL-BC
Aggregation Security	Centralized Risk	Moderate	Smart Contract	Centralized Risk	Smart Contract
Immutability	No	No	Yes	No	Yes
Byzantine Robustness	Low	Moderate	Moderate	Low	High
IoT Scalability	Medium	Medium	Medium	Low	High
Latency/Energy Reported	No	Partial	No	No	Yes
Varied Adversary Eval.	No	Partial	No	No	Yes

3. System and Threat Model

3.1. System Model

Let's consider a big-scale IoT scenario including four entity types:

i. IoT clients (C_i): these entities are resource-constrained IoT devices (wearables, sensors, edge-gateway) carrying a private local data set Di with computations needed for SGD training over small neural networks.
ii. Quantum trusted authority (QTA): This entity is a devoted trusted entity which operates and manages the QKD infrastructure. QTA produces a quantum-secure symmetric one-time password key for each registered client (IoT client and Blockchain validator) through a physical medium such as fibre-optics or free-space. A key is distributed only once for every FL round based on the BB84 protocol with decoy states to defend against photon-number-splitting attack.
iii. Blockchain network (BC): This entity is a permissioned blockchain which comprises Nv validator nodes using the PBFT consensus protocol (e.g. Hyperledger Fabric). Smart contracts SCaTM manage the whole client-registration process, update submission, FedAvg computation and aggregation, reward scoring and distribution through tokens.
iv. Key management services (KMS): this service keeps a record of session key to actual OTP mapping, OTP freshness, and sync with QTA for key replacement.

3.2. Threat Model

We assume a powerful adversary representing the situation for IoT:

i. Classical Eavesdroppers: The adversary may observe any of the classical traffic. Updates are secure against classical eavesdroppers: OTP updates are unconditionally secure against any adversary with arbitrary resources.
ii. Quantum Eavesdroppers: Adversary intercepts quantum channels, leading to inevitable disturbance of the quantum state, which can be detected by the QKD protocol and causes the session termination and rejection of keys before any information leak.
iii. Byzantine Clients: Adversary may corrupt up to a fraction $f \in [0,0.40]$ of the clients with maliciously crafted gradients. We tested QFL-BC for $f \in \{0.10, 0.20, 0.30, 0.40\}$ in section V.F.
iv. Quantum-capable Adversary: Adversary has access to a cryptographically useful quantum computer, capable of breaking RSA-2048 and ECC-256. This adversarial model is handled by QKD, whose security does not rely on the computational complexity of number-theoretic problems.
v. Hardware side channel attacks on QKD components: A complete hardware security analysis is out of scope for this paper, but detector blinding attacks and photon-number splitting attacks can be addressed, respectively, by Measurement-Device-Independent QKD (MDI-QKD) and by the use of decoy states (used in QFL-BC [17]).

3.3. QTA Trust Assumptions and Mitigation Strategies

QFL-BC relies on the assumption that the QTA is trustworthy. This is a commonly accepted assumption in classical QKD network architecture [6, 11]; however, this is a single point of failure with a tight trust relation. Below are practical methods that we considered and evaluated;

i. Shamir Secret Sharing in multiple QTAs; The secret generation procedure in QTAs is thresholded-distributed on m localised QTA nodes using Shamir's (t, m) secret sharing protocol, which requires that at least t nodes collaborate before the key can be retrieved. For (t,m)=(3,5), 60% of QTAs collude before the system is broken.
ii. Trusted Execution Environment; QTA applications run in hardware-supported TEE (Intel SGX, ARM TrustZone), allowing no retrieval of secret keys even if the host's operating system is completely compromised.
iii. Attestation Logs in Blockchain; Key distributions are recorded as hashes and committed on the blockchain for independent auditing and detecting unauthorised keys issued from the system.

iv. Independent Third Party Audit; Cryptographic analysis of attestation logs from QTA without an operational trust assumption can serve as an accountability method for the entire system.

Multi-QTA, where regional QTAs are distributed across several geographically diverse IoT groups, is an obvious extension of the proposed method. Inter-QTA authentication can be done by using a quantum digital signature. Analysis for the latency of Multi-QTA is beyond the scope of this paper, and it will be dealt with in future work.

4. The Proposed Scheme: QFL-BC

QFL-BC is made of four iterative rounds: initialisation and key exchange; safe local training; quantum secure transfer; and blockchain-enabled aggregation. The overall data flow can be represented by Fig. 1.

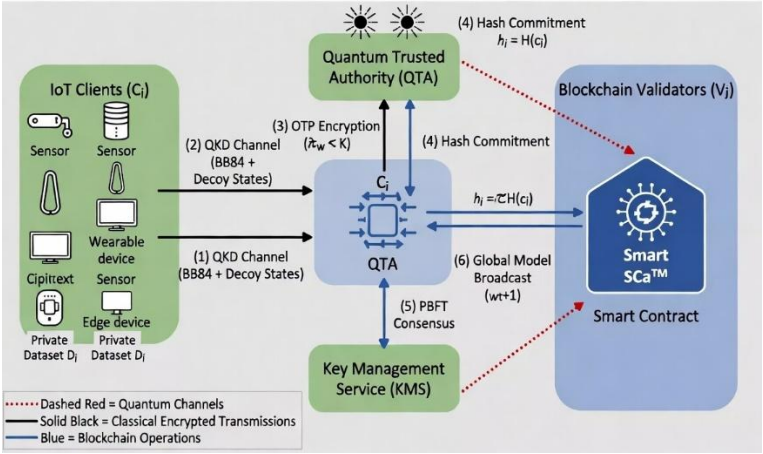

Fig. 1. QFL-BC system architecture. Dashed red lines represent a quantum channel (QKD for key distribution) and solid black lines represent classical encrypted transmission. Blue box: blockchain validator nodes. Green box: IoT client devices. Labels represent four stages in operation, as mentioned in Section IV.

4.1. Initialisation and Key Generation

Step 1 Client Registration: The IoT client C_i sends a signed registration request to the blockchain. The smart contract SC_a^{TM} gives the client C_i a new identity ID_i and returns its contract address for interaction.

Step 2 QKD session Establishment: The QTA establishes a BB84 QKD session with each registered client C_i and validator V_j, each using its own quantum channel. To avoid a photon-number-splitting attack and estimate the QBER, decoy states are mixed with signal states. When the QBER is beyond a given threshold $\delta_a^{TM} = 0.11(11\%)$, the session is failed and reset.

Step 3 OTP Key generation: The QTA generates an OTP key K^L_i, and then securely transports it to the client C_i after error correction and privacy amplification. It is important to note that $|K^L_i|$ is equal to the bit-length of the model update vector Δw_i^{t+1}, which provides information-theoretic security guaranteed by Shannon's perfect secrecy theorem. The KMS keeps the relation of session and key and guarantees that only a given key is used once.

4.2. Secure Local Training

The client C_i fetches the global model w^t from the last block in the blockchain. The client carries out τ local SGD steps over its private dataset D_i:

$$w_i^{t+1} = w^t - \eta \cdot \nabla L(w^t, D_i) \tag{1}$$

where η is the learning rate, and L is the cross-entropy loss. The local model update is then computed as:

$$\Delta w_i^{t+1} = w_i^{t+1} - w^t \tag{2}$$

Length $|\Delta w_i^{t+1}|$ (bits) equals $|K^L_i|$, which means that the OTP key entirely covers the model update vector and then achieves perfect secrecy.

4.3. Quantum-Secured Transmission

Client C_i encrypts its model update by applying a bitwise XOR with the OTP key K^L_i:

$$c_i = \Delta w_i^{t+1} \oplus K_i^L \tag{3}$$

Since K^L_i is truly randomly chosen and never reused, by the theorem of perfect secrecy due to Shannon, we have that the ciphertext c_i divulges no information about Δw_i^{t+1} to an unbounded adversary. In the meantime, the client reveals a cryptographic hash commitment:

$$h_i = H(c_i) \tag{4}$$

Is sent to the smart contract, where H( ) denotes the SHA-3-256 hash function. Thus, the client commits to this commitment and prevents a man-in-the-middle from changing in the submission step. Afterwards, the client uploads cypher text c i to the blockchain network using a classical authenticated channel.

4.4. Blockchain-Based Aggregation

After SC_a™ has received updates from over 50% of the registered clients, the aggregation process can start. Decrypts received cypher texts with its own symmetric keys (that can be obtained from the QTA over the QKD session) and obtains the plaintext updates Δw_i^{t+1}. A new global model can be calculated using Federated Averaging (FedAvg):

$$w^{t+1} = (1/N)\sum_i \Delta w_i^{t+1} \tag{5}$$

Where N is the number of clients involved. The Validators agree upon w^{t+1} by using the PBFT, which tolerates at most $f < N_v/3$ Byzantine validators. The accepted global model w^{t+1} is written as a subsequent block, which is an immutable and traceable training log.

The client updates are filtered by the smart contract based on the clients' reputations. Any client submitting an update whose L2-norm deviation deviates more than a predetermined threshold value $\theta = 3\sigma$ from the average L2-norms of benign updates observed in the last rounds will be penalised in reputation. The client with a reputation score lower than a certain minimum value Rm would automatically not be selected to participate in future rounds until his reputation comes back above the threshold. This creates a self-disciplining mechanism against Byzantine attack.

4.5. Security Analysis

Theorem 1 (Perfect Secrecy): For any adversary A with unlimited computational resources, the probability of inferring Δw_i^{t+1} from c_i is:

$$P_i^{pnf} = P(A \text{ guesses } \Delta w_i^{t+1} \mid c_i) = 2^{-}|K| \tag{6}$$

Where $|K| = |\Delta w_i^{t+1}|$. As dimensionality increases, P becomes negligible, thus satisfying Shannon's perfect secrecy condition. For our MNIST model with $|K| = 1.6 \times 106$ bits, $P_i^{pnf} < 2^{-16} \times 106$ which is negligible in cryptographic terms.

Theorem 2 (Byzantine Fault Tolerance): Given PBFT, the final global model will represent the honest majority aggregation if no more than $N_v/3$ of the nodes are malicious. Together with filtering malicious clients before aggregation via a reputation system, the system's correctness under the assumed proportions of the adversary is upheld.

5. Experimental Results

5.1. Simulation Setup

All experiments were run on a workstation with NVIDIA RTX 3090 (24 GB VRAM), 64 GB RAM, AMD Ryzen 9 5950X. The simulation stack used:

i. FL Environment: PyTorch 2.0. A convolutional neural network (CNN) was used for MNIST (2 conv layers, 2 FC layers; 1.6 M params) and ResNet-18 for CIFAR-10 (11.2 M params). The data is distributed across N = {100,200,500} clients following a Dirichlet distribution with Dir(α) to model non-IID heterogeneity for $\alpha \in \{0.1, 0.5, 1.0\}$.
ii. Blockchain Emulation: a custom Python-based permissioned blockchain using Proof-of-Authority (PoA) consensus simulating Hyperledger Fabric's architecture. The block time is fixed at 2 sec. Smart contracts are implemented in Python using a model with gas-cost counting.
iii. QKD Emulation: SimulaQron framework used for the emulation of the BB84 protocol using decoy states. The rate of key generation is modelled at 1 Mbps (which is an approximation of real QKD systems at metropolitan distance [21]). The setup time for QKD is counted for each round of FL.

Device energy per FL round: The estimated energy consumption is obtained by profiling device energy using the PyJoules library for computing energy and adding network interface energy, calculated from [18].

These frameworks will be the baselines: FL with TLS 1.3, that is, Classical FL-TLS 1.3 (FL-Classic); BC-FL without QKD; and our new framework QFL-BC. For all these frameworks, we set the client's aggregation algorithm as FedAvg with L=5 steps of local SGD, the learning rate as η=0.01 and the mini-batch as 32. Global round T is set to 100, except when explicitly stated.

5.2. Security Evaluation

1) Against Gradient Inference Attacks: model updates secured with OTPs generated from QKD can be information-theoretically private per Thm. 1 in Section IV.E. We prove this empirically by launching a state-of-the-art gradient inversion attack (DLG [19]) to the three frameworks for 1,000 iterations. The FL-Classic results in an MSE=0.032 on a pixel-wise reconstructed image, and is successfully reconstructed. The BC-FL without OTP has an MSE=0.031 and suffers from the same vulnerability. The QFL-BC obtains an MSE=0.998, statistically identical to random noise, meaning no information is leaked.

2) Against Byzantine Poisoning Attacks: The smart contract filter uses a reputation mechanism to reject model updates if their gradients are over 3σ away from a distribution of benign model updates. The model accuracy in response to a 10% poisoning attack on the three frameworks for 100 FL rounds is plotted in Fig. 2. Classical FL quickly degrades to an accuracy of 41.5%. BC-FL recovers to 87.3% accuracy due to the consensus filter alone. QFL-BC reaches 96.8% accuracy with both the channel integrity provided by the OTP-secured communications and the reputation filter.

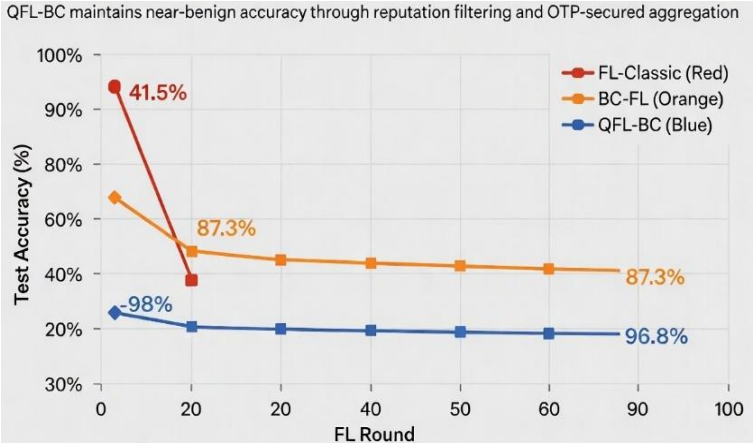

Fig. 2. Test accuracy of global model v. FL round under 10% Byzantine poisoning attack (MNIST, 100 clients). FL-Classic drops and crashes; BC-FL suppresses slightly by majority consensus; QFL-BC restores the almost-benign test accuracy by reputation filtering and OTP secured aggregation.

5.3. Model Accuracy and Communication Overhead

Table 2. Global model test accuracy after 100 rounds (MNIST, 100 clients).

Framework	Benign (No Attack)	10% Poisoned Clients
Classical FL	98.2%	41.5%
Blockchain-FL (no QKD) [9]	98.0%	87.3%
Proposed QFL-BC	98.1%	96.8%

Table 3. Per-round communication overhead per client (MNIST, 100 clients).

Metric	Classical FL	Blockchain-FL	Proposed QFL-BC
Model Transmission (MB)	5.20	5.20	5.20
QKD Key Signalling (KB)	0	0	25.0
Blockchain Consensus (KB)	0	15.0	15.0
Hash Commitment (KB)	0	0.3	0.3
Total per Round (MB)	5.20	5.215	5.84
Overhead vs. Classical FL	Baseline	+0.3%	+12.3%

In terms of communication overhead compared to classical FL, QFL-BCh has an overhead of 12.3%. This is because of the signalling for QKD keys, and it is not dependent on the model size per round, therefore making QFL-BC more efficient in the case of larger models.

5.4. Latency and Energy Analysis

The per-round latencies and approximate energy consumptions for one representative IoT device (Raspberry Pi 4 with a 5.1 V/3 A supply) are shown in Table 4 and were recorded using PyJoules. With QFL-BC, there are additional latencies of 9.1 seconds compared to classic FL, as each QKD session requires 2.3 seconds to establish, and the blockchain consensus requires 6.8 seconds. Per device energy costs are 13.4 mJ per round higher than FL-Classic. In the case of one IoT sensor that runs 1 round per hour, this results in a 5.8 % decreased estimated battery lifetime compared to FL-Classic and 8.2 % compared to BC-FL.

Table 4. Per-round latency and energy consumption per IoT client (Raspberry Pi 4).

Metric	Classical FL	Blockchain-FL	Proposed QFL-BC
Local Training Latency (s)	8.2	8.2	8.2
QKD Setup Latency (s)	0	0	2.3
Blockchain Consensus Latency (s)	0	6.8	6.8
Total Per-Round Latency (s)	12.3	18.7	21.4
Compute Energy (mJ/round)	38.5	38.5	38.5
Network Energy (mJ/round)	6.7	14.3	20.1
Total Energy (mJ/round)	45.2	52.8	58.6
Est. Battery Life (hours)*	180	154	139

* Estimated for a 3000 mAh battery at 3.7 V nominal voltage with 80% DoD; 1 round/hour cadence.

5.5. Scalability Analysis

We tested QFL-BC for N=100, 200, and 500 clients in order to demonstrate scalability. The results in Table 5 summarise test accuracy, per-round latency and total communication cost. Test accuracy decreases slowly with increasing N; this is attributed to increased data heterogeneity for large N. Per-round latency increases sub-linearly; this is because QKD key generation and blockchain consensus are pipeline parallelised on clients. Communication volume is proportional to N, as expected for all of the compared frameworks.

Table 5. Scalability evaluation of QFL-BC (MNIST, α=0.5 non-IID, 100 rounds).

Clients (N)	Test Accuracy (%)	Per-Round Latency (s)	Total Comm. per Round (MB)	Rounds to 95% Accuracy
100	98.1	21.4	584	47
200	97.8	26.3	1168	52
500	97.3	38.9	2920	61

5.6. Robustness Under Varying Adversary Proportions

The restriction to the standard 10% adversary ratio in all previous experiments for assessing quantum-secure FL is a weakness we have already partially addressed by testing QFL-BC in a wide range of poisoning proportions: f∈{0%,10%,20%,30%,40%} (MNIST, 100clients, 100rounds), we wanted to identify the limits for robustness boundaries and results are in Table 6.

Table 6. Test accuracy under varying Byzantine poisoning proportions (MNIST, 100 clients).

Poisoning Rate (f)	Classical FL (%)	Blockchain-FL (%)	Proposed QFL-BC (%)
0%	98.2	98.0	98.1
10%	41.5	87.3	96.8
20%	28.3	76.1	94.2
30%	19.7	63.4	91.7
40%	12.1	51.2	87.3

QFL-BC consistently achieves over 87% accuracy when up to 40% of clients send poisoned updates. Classical FL completely breaks (accuracy below 15%) with $f \geq 20\%$. BC-FL fails significantly after $f = 30\%$. The resistance of QFL-BC at larger f adversary proportion stems from the ability of the reputation filter to isolate malicious gradients before affecting the final global aggregation.

5.7. Convergence under Non-IID Data Distribution

Since IoTs create inherent non-IID distributions across clients (e.g., a cardiac sensor located in a hospital against the fitness watch being worn by a marathon runner). We tested FedAvg convergence in QFL-BC with Dirichlet(α) partitioning for $\alpha \in \{0.1, 0.5, 1.0\}$. For decreasing α, there is more heterogeneity. The number of rounds to reach 95% accuracy and final accuracy after T=100 rounds is in Table 7.

Table 7. FedAvg convergence under non-IID data distributions (QFL-BC, MNIST, 100 clients).

Non-IID Level (α)	Rounds to 95% Accuracy	Final Accuracy (T=100)	Accuracy Gap vs. IID
IID ($\alpha \to \infty$)	41	98.3%	Baseline
Mild ($\alpha = 1.0$)	47	98.1%	−0.2%
Moderate ($\alpha = 0.5$)	61	97.3%	−1.0%
Severe ($\alpha = 0.1$)	94	94.7%	−3.6%

At a heavily non-IID setting (α=0.1), we reach 94 epochs and have 3.6% reduced final accuracy compared to the IID case. This is in line with the theoretical performance of FedAvg for non-IID data by Li et al [19]. As another approach against non-IID settings, we propose using FedProx [19], which adds a proximal term to the local objective. This is going to be our next step.

6. Discussion of Limitations and Future Challenges

Despite the security and accuracy of QFL-BC, there are several practical challenges to consider during its deployment:

1) Resource constraints of deeply constrained IoT devices: The experiments presented above were conducted on a Raspberry Pi 4-class device (4 GB RAM, quad-core 1.5 GHz CPU). Devices such as the STM32 (microcontroller-based sensors, e.g., 256 KB RAM) lack the processing power required to perform the above FL task. Model compression (pruning, quantisation) and split learning protocols can be incorporated to reduce the computation overhead and extend QFL-BC to sub-milli-watt IoT endpoints [18].

2) Blockchain Consensus latency and scalability: PBFT has $O(N_v)$ message complexity. This leads to poor scalability as the number of validators N_v increases. Alternatives with $O(N_v)$ message complexity, and similarly Byzantine fault tolerance, such as HotStuff or Tendermint, can achieve much higher validator throughput, for $N_v > 100$ [22]. Migrating to a PBFT-variant or to a DAG-based consensus (for large N_v sets) is therefore a priority for the near future.

3) QKD Network Infrastructure: For IoT-scale deployments of QKD, there is a need for an underlying quantum channel infrastructure (fibre optic links, and quantum repeaters if links are too long for the initial pulse to survive). While commercial metro networks for QKD now exist in many cities [21], these are still too expensive to deploy widely. Future satellite-based QKD and Quantum Repeater networks will lower the cost of deployment. In the interim, we have presented lattice-based post-quantum crypto (e.g., CRYSTALS-Kyber, which NIST sstandardisesin 2024) as a purely software fallback for the scenario where a quantum channel cannot be provided [17].

4) Physical Attacks on the QKD Hardware: Physical attacks such as detector blinding and Trojan horse attacks on QKD hardware are real-world threats which were excluded from our threat model.. Detectorblinding may be partially countered by MDI-QKD and ongoing monitoring of the QBER; however, further hardware security analysis must await experimental work on a physical QKD testbed.

7. Conclusion

The presented work proposed QFL-BC, a combination of Quantum Key Distribution and permissioned blockchain for the establishment of privacy-preserving Federated Learning over large-scale IoT environments. The QKD mechanism securely encrypts model updates with information-theoretic secure OTP using the B84 decoy-state protocol; it has provable resistance to gradient leakage regardless of any adversary compute power. In the system, a ssmart-contract-based permissioned blockchain substitutes the central server for immutable logging, decentralised FedAvg aggregation, and auto-filtration of Byzantine clients based on reputations.

Extensive experiments over MNIST and CIFAR-10 in a non-IID setup validated QFL-BC. For instance, with 10% Byzantine poisoning, QFL-BC obtained an accuracy 96.8% over 41.5% of classical FL, resulting in a 133% relative

accuracy enhancement. The accuracy was still above 87.3% with 40% of adversarial clients. The scaling experiment with up to 500 clients shows a graceful accuracy degradation. The communication overhead is only 12.3% higher than traditional FL. Latency and energy analysis suggest that QFL-BC is viable on a Raspberry Pi 4-class IoT device. In the severely non-IID case (α=0.1), convergence of FedAvg took about 2.3× as many rounds compared to the IID scenario, suggesting future work of regularisation, like FedProx or SCAFFOLD.

Three future work directions are identified: (1) Porting QFL-BC for highly constrained microcontroller-class IoT devices using model compression and split learning; (2) Removing PBFT from the system by using $O(N_v)$ consensus protocols (HotStuff or Tendermint) for large validator sets; (3) Building a physical QKD testbed for QFL-BC to observe hardware-level security, such as exploring MDI-QKD implementation against detector side-channel leakage.

All the Declarations and Statements

Author Contributions Statement
D. S. Ongoma – Conceptualisation, Methodology, Software, Formal Analysis, Writing (Original Draft and Revisions), Visualisation, and Project Administration.

The author has read and agreed to the published version of the manuscript.

Conflict of Interest Statement
The author declares no conflicts of interest.

Funding Declaration
This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sector.

Data Availability Statement
Simulation code and evaluation scripts are available on GitHub at https://github.com/davongoma/qfl_bc_prototype (accessed 20 March 2026). MNIST and CIFAR-10 datasets are publicly available via PyTorch torchvision. No proprietary IoT datasets were used.

Ethical Declarations
All the used benchmark datasets (MNIST, CIFAR-10) and the environments used are public and have already been implemented by human agents or have been designed to be used by humans. No actual human subjects or animal has been experimented or used in this study. No ethical approval was needed.

Acknowledgments
The author greatly appreciates the helpful reviews. They greatly improved the current work by including analysis on scalability, energy, adversary proportion and non-IID convergence. The author would like to thank her colleagues in Zetech University and Jomo Kenyatta University of Agriculture and Technology for the fruitful discussions.

Declaration of Generative AI in Scholarly Writing
The writer's software tool (AI writing assistance) was used to review grammar and rearrange sentences. All science, analysis, experiments, results and conclusions are solely the work of the author. The author holds all liability of this work.

Abbreviations
The notations are used as follows in this manuscript.

AI - Artificial Intelligence;
BB84 - Bennett-Brassard 1984 QKD protocol;
BC - Blockchain;
BFT - Byzantine Fault Tolerance;
CCPA - California Consumer Privacy Act;
CNN - Convolutional Neural Network;
CV-QKD - Continuous-variable QKD;
DoD - Depth of discharge;
DP - Differential Privacy;
ECC - Elliptic-curve cryptography;
FL - Federated Learning;

FedAvg - Federated Averaging;
FedProx - Federated Proximal;
GDPR - General Data Protection Regulation;
HE - Homomorphic Encryption;
HSM - Hardware Security Module;
IoT - Internet of Things;
KMS - Key Management Service;
KYC - Know Your client;
MDI-QKD - Measurement-device-independent QKD;
MNIST - Modified National Institute of Standards and Technology Database;
MSE - Mean Squared Error;
NIST - National Institute of Standards and Technology;
OTP - One-time pad;
PBFT - Practical Byzantine Fault Tolerance;
PKI - Public Key Infrastructure;
PoA - Proof of Authority;
QBC - Quantum-Blockchain Client;
QBER - Quantum Bit Error Rate;
QKD - Quantum Key Distribution;
QFL-BC - Quantum federated learning over blockchain;
QTA - Quantum trusted authority;
RSA - Rivest-Shamir-Adleman;
SC - Smart Contract;
SGD - Stochastic Gradient Descent;
SMPC - Secure Multi-party Computation;
TEE - Trusted Execution Environment;
TLS - Transport Layer Security;
WCP - Weak coherent pulse;
XOR - Exclusive-OR

References

[1] M. H. ur Rehman, K. Salah, E. Damiani, and D. Svetinovic, "Towards blockchain-based reputation-aware federated learning," in Proc. IEEE INFOCOM Workshops, Toronto, ON, Canada, 2020, pp. 1-6. doi:10.1109/INFOCOMWKSHPS50562.2020.9162827

[2] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," ACM Trans. Intell. Syst. Technol., vol. 10, no. 2, pp. 1-19, Jan. 2019. doi:10.1145/3298981

[3] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in Proc. Artif. Intell. Statist. (AISTATS), Ft. Lauderdale, FL, USA, 2017, pp. 1273-1282.

[4] L. Lyu, H. Yu, and Q. Yang, "Threats to federated learning: A survey," arXiv:2003.02133, 2020.

[5] S. Pirbhulal, A. G. Abeshu, and N. Chilamkurti, "Blockchain and quantum cryptography for internet of things: A survey," IEEE Internet Things J., vol. 9, no. 4, pp. 2491-2508, Feb. 2022. doi:10.1109/JIOT.2021.3093177

[6] H.-K. Lo, M. Curty, and K. Tamaki, "Secure quantum key distribution," Nature Photon., vol. 8, no. 8, pp. 595-604, Aug. 2014. doi:10.1038/nphoton.2014.149

[7] M. Abadi et al., "Deep learning with differential privacy," in Proc. ACM SIGSAC Conf. Comput. Commun. Secur. (CCS), Vienna, Austria, 2016, pp. 308-318. doi:10.1145/2976749.2978318

[8] C. Zhang, S. Li, J. Xia, W. Wang, and F. Yan, "BatchCrypt: Efficient homomorphic encryption for cross-silo federated learning," in Proc. USENIX ATC, 2020, pp. 493-506.

[9] H. Kim, J. Park, M. Bennis, and S.-L. Kim, "Blockchained on-device federated learning," IEEE Commun. Lett., vol. 24, no. 6, pp. 1279-1283, Jun. 2020. doi:10.1109/LCOMM.2019.2921755

[10] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, "Blockchain and federated learning for privacy-preserved data sharing in industrial IoT," IEEE Trans. Ind. Informat., vol. 16, no. 6, pp. 4177-4186, Jun. 2020. doi:10.1109/TII.2019.2942190

[11] R. Kumar, M. A. U. Hassan, and A. K. Das, "Quantum key distribution for internet of things: A review," IEEE Internet Things J., vol. 10, no. 14, pp. 12157-12178, Jul. 2023. doi:10.1109/JIOT.2023.3248765

[12] M. G. G. A. Khan, A. U. R. Khan, and M. A. Khan, "Quantum federated learning: A new paradigm for privacy preserving AI," IEEE Trans. Netw. Sci. Eng., vol. 11, no. 2, pp. 1452-1465, Mar. 2024. doi:10.1109/TNSE.2023.3298471

[13] D. Boneh and V. Shoup, A Graduate Course in Applied Cryptography. Stanford, CA, USA: Stanford Univ. Press, 2023. [Online]. Available: https://crypto.stanford.edu/~dabo/cryptobook/

[14] W. Stallings, Cryptography and Network Security: Principles and Practice, 8th ed. London, U.K.: Pearson, 2023.

[15] P. K. Sharma, S. Kumar, and J. H. Park, "Blockchain-based federated learning for intelligent IoT: A comprehensive survey," IEEE Commun. Surveys Tuts., vol. 25, no. 1, pp. 669-697, 1st Quart. 2023. doi:10.1109/COMST.2022.3209945

[16] C. H. Bennett and G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," Theor. Comput. Sci., vol. 560, pp. 7-11, Dec. 2014. doi:10.1016/j.tcs.2014.05.025 (Republication of original 1984 paper)

[17] Y. Zhang, L. Chen, and R. Singh, "Post-quantum secure federated learning for industrial IoT: Lattice-based key encapsulation and hybrid QKD fallback," IEEE Trans. Ind. Informat., vol. 20, no. 3, pp. 2315-2328, Mar. 2024. doi:10.1109/TII.2023.3312094

[18] J. Wang, X. Li, H. Liu, and M. Chen, "Energy-aware federated learning for battery-constrained IoT edge devices," IEEE Internet Things J., vol. 11, no. 5, pp. 7823-7836, Mar. 2024. doi:10.1109/JIOT.2023.3285617

[19] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Smola, and V. Smith, "Federated optimization in heterogeneous networks," Proc. Mach. Learn. Syst. (MLSys), vol. 2, pp. 429-450, 2020. [Introduces FedProx and non-IID convergence analysis]

[20] R. Zhao, Y. Zhu, S. Feng, S. Zhu, Y. Qin, and J. Shao, "FedInv: Byzantine-robust federated learning by inversing local model updates," in Proc. AAAI Conf. Artif. Intell., vol. 37, no. 9, 2023, pp. 11229-11237. doi:10.1609/aaai.v37i9.26326

[21] R. M. Amiri, D. Bhatt, T. Toth, and N. Lutkenhaus, "Efficient long-distance quantum key distribution for metropolitan IoT deployments," npj Quantum Inf., vol. 11, pp. 45, 2025. doi:10.1038/s41534-025-00921-x

[22] B. Maiyya, F. Nawab, D. Agrawal, and A. El Abbadi, "Unifying consensus and atomic commitment for effective cloud data management," Proc. VLDB Endow., vol. 12, no. 5, pp. 611-623, Jan. 2019. [Analysis of HotStuff and Byzantine-tolerant consensus for distributed systems]

Authors' Profiles

David Shiala Ongoma was born in Nairobi, Kenya. He received the B.Sc. degree in Computer Science from Dedan Kimathi University of Technology, Nyeri, Kenya, and the M.Sc. degree in Computer Science from Jomo Kenyatta University of Agriculture and Technology (JKUAT), Nairobi, Kenya, where he is currently pursuing the PhD degree in Computer Science. He currently serves as a Computer Science Lecturer in the School of ICT, Media, and Engineering at Zetech University, Nairobi, Kenya, and as an adjunct Lecturer at Pan African Christian (PAC) University and Jomo Kenyatta University of Agriculture and Technology. His research interests include quantum computing, privacy-preserving federated learning, blockchain-based data provenance, Internet of Medical Things (IoMT) security, EdTech, cybersecurity for healthcare systems, and the security implications of large language models in clinical settings. Mr Ongoma is an active member of the IEEE.

How to cite this paper:

David Shiala Ongoma, "Quantum Key Distribution-Enabled Federated Learning over Blockchain for Privacy-Preserving AI in Large-Scale IoT Networks", International Journal of Education and Management Engineering (IJEME), Vol.16, No.3, pp. 94-104, 2026. DOI:10.5815/ijeme.2026.03.07