World Journal of Advanced Research and Reviews

eISSN: 2581-9615 CODEN (USA): WJARAI

Cross Ref DOI: 10.30574/wjarr

Journal homepage: https://wjarr.com/

(Research Article)

Quantum-AI Federated Clouds: A trust-aware framework for cross-domain observability and security

Omoniyi David Olufemi *

Department of Computer Engineering, University of Fairfax, VA, United States.

World Journal of Advanced Research and Reviews, 2025, 26(02), 4098–4140

Publication history: Received on 16 April 2025; revised on 25 May 2025; accepted on 27 May 2025

Article DOI: https://doi.org/10.30574/wjarr.2025.26.2.2074

Abstract

The convergence of quantum computing, artificial intelligence (AI), and federated cloud architecture offers transformative potential for secure, scalable, and privacy-preserving data processing. Yet, trust management and cross-domain observability remain major challenges, particularly in decentralized, heterogeneous cloud environments. This paper introduces Quantum-AI Federated Clouds (QAIFC) a novel trust-aware framework that combines quantum-safe encryption, federated machine learning, and explainable AI to enable secure and observable operations across cloud domains. We present QFedSecure, a protocol suite leveraging lattice-based cryptography, quantum key distribution, and AI-driven anomaly detection to support trust propagation and policy enforcement. The framework features a dynamic trust model, observability protocol, and mechanisms for adversarial resilience. Simulations using Qiskit, TensorFlow Federated, and NS3 show up to 40% improvement in trust calibration and 55% increase in adversarial detection over baseline systems. This work advances the foundation for resilient, decentralized, and quantum-secure AI cloud ecosystems.

Keywords: Post-Quantum Encryption; Quantum Key Distribution (QKD); Zero-Knowledge Proofs (ZKPs); Federated Learning (FL); Explainable AI (XAI); Anomaly Detection in FL; Dynamic Trust Scoring; Differential Privacy (DP); Zero Trust Architecture (ZTA); Cross-Domain Observability

1. Introduction and Problem Statement

The ongoing transformation of digital infrastructure fueled by artificial intelligence (AI), ubiquitous cloud services, and the early-stage development of quantum computing is pushing the boundaries of conventional security, data privacy, and computational coordination. Federated learning (FL) has emerged as a scalable alternative to centralized AI by enabling decentralized training across multiple participants without the need for raw data exchange (Kairouz et al., 2021). In this architecture, models are trained locally and aggregated globally, preserving data privacy and regulatory compliance.

However, federated learning assumes a baseline of trustworthiness among participants and often lacks visibility across administrative domains, making it vulnerable to attacks and inconsistencies in policy enforcement. In addition, the rapid progression of quantum computing through advances in superconducting qubits, trapped ions, and photonic processors poses an existential threat to classical cryptographic systems used in federated cloud infrastructures (Shor, 1994). As a result, the need for a unified framework that combines quantum-safe protocols, trust awareness, and AI observability has become both timely and critical.

* Corresponding author: Omoniyi David Olufemi

1.1. Motivation for Quantum-AI Federated Cloud Systems

Federated systems are increasingly distributed across national boundaries and diverse cloud service providers (CSPs), making issues of interoperability, observability, and security more complex. The motivation to build Quantum-AI Federated Clouds (QAIFC) stems from the following:

- **Rising threats to classical encryption:** Quantum computing algorithms such as Shor's and Grover's can compromise RSA, ECC, and symmetric encryption, necessitating post-quantum cryptographic measures (Bernstein et al., 2017).
- **Lack of global trust anchors:** Cross-domain federated systems often operate without centralized authorities or shared Public Key Infrastructures (PKIs), leading to inconsistencies in identity management and trust verification.
- **Opacity in AI model updates:** Malicious actors may manipulate gradients or inject poisoned updates in federated learning environments, with limited explainability tools available to detect or diagnose such behavior (Bhagoji et al., 2019).
- **Lack of end-to-end observability:** Many federated systems fail to track data provenance, resource usage, and threat response in real time, especially across federated boundaries (Zhou et al., 2021).

These gaps highlight the urgent need for a trust-aware, quantum-resilient framework capable of integrating federated AI with security observability and trust management mechanisms.

1.2. Core Research Problem

The primary research problem addressed in this paper is: How can federated cloud systems achieve secure, observable, and trust-aware AI collaboration across quantum-threatened, cross-domain infrastructures without centralized control?

This problem is complex due to the following intertwined challenges:

Federated cloud systems face the critical challenge of enabling secure, observable, and trust-aware AI collaboration across quantum-threatened, cross-domain infrastructures without relying on centralized control. This complexity arises from several intertwined factors: trust is inherently dynamic and multi-dimensional, requiring continuous evaluation based on behavioral indicators, policy compliance, and cryptographic verification; observability is often fragmented, with each domain operating isolated monitoring systems that create exploitable blind spots; security mechanisms must evolve to withstand future quantum threats, necessitating a shift from classical to quantum-resistant cryptographic protocols; and finally, AI models must be not only accurate but also explainable and resilient, capable of withstanding adversarial manipulation while maintaining interpretability across federated environments.

1.3. Objectives of the Study

This paper proposes a layered architectural framework and protocol suite QFedSecure designed to meet several critical objectives in securing federated AI systems. First, it integrates Post-Quantum Cryptography (PQC) into the federated learning pipeline, incorporating lattice-based encryption, ring-LWE, and Quantum Key Distribution (QKD) to ensure future-proof security (Alkim et al., 2016; Pirandola et al., 2020). Second, it introduces a cross-domain trust model, providing a mathematical framework for computing and dynamically updating trust scores based on behavioral metrics, cryptographic attestations, and explainable AI feedback. Third, the framework includes a Cross-Domain Observability Protocol (CDOP) to facilitate real-time monitoring of data exchanges, model updates, and anomaly detection across federated nodes. Additionally, Explainable AI (XAI) modules are deployed to interpret model decisions, detect poisoning attempts, and enhance accountability in AI workflows (Ghosh et al., 2022). Finally, the system's robustness is validated through simulations using tools such as Qiskit, TensorFlow Federated, and NS3, evaluating its performance under adversarial and quantum-resilient conditions.

1.4. Contributions of the Study

This research makes significant contributions to both theory and practice by introducing a novel trust-aware federated cloud architecture that seamlessly integrates quantum-safe cryptography and explainable AI to enable secure, end-to-end collaboration across distributed systems. At its core is the QFedSecure protocol suite, which facilitates encrypted gradient exchange, dynamic trust scoring, and federated orchestration across untrusted or semi-trusted domains. The framework also incorporates a robust observability layer, leveraging AI-driven anomaly detection and comprehensive audit logging to ensure traceability and verification of all model-related activities. Furthermore, the study advances mathematical modeling and algorithmic design, offering formal trust scoring equations, privacy guarantees, and cryptographic resilience under both classical and quantum threat models. These innovations are supported by a rigorous validation framework that measures key performance indicators including trust propagation, anomaly detection accuracy, communication overhead, and model degradation in adversarial environments.

1.5. Scope and Limitations

The proposed framework focuses on cross-domain federated learning environments, where each domain may be a sovereign cloud provider or edge node cluster. While the architecture integrates quantum-safe protocols, it assumes access to quantum-secure communication channels (e.g., QKD networks) for key distribution. Moreover, the XAI modules emphasize interpretability over deep learning black-box optimization, which may trade off performance in certain contexts. Lastly, real-world implementation challenges such as network heterogeneity, jurisdictional restrictions, and economic incentives are acknowledged but not exhaustively addressed in this study.

2. Literature Review and Theoretical Foundation

2.1. Federated Learning and Decentralized AI Orchestration

Federated Learning (FL) is a decentralized machine learning technique that enables the training of global models using local datasets on client devices without transferring data to a centralized server. Originally introduced by McMahan et al. (2017), the FL paradigm has evolved into a cornerstone for privacy-preserving AI, particularly within healthcare, finance, and cross-organizational applications.

2.2. Federated Learning and Decentralized AI Orchestration

Federated Learning (FL) was introduced to address the growing need for collaborative model training across distributed data silos without requiring the centralization of sensitive data (McMahan et al., 2017; Kairouz et al., 2021). In conventional machine learning paradigms, data must be uploaded to a central server where models are trained. This model poses significant risks related to data privacy, compliance with regulations like GDPR and HIPAA, and scalability issues when dealing with massive, decentralized datasets.

FL circumvents these issues by allowing each participating node (e.g., edge devices, cloud data centers, mobile clients) to locally train a copy of the global model using its private data. These updates are then aggregated typically on a central server or distributed coordinator without transferring raw data. This approach upholds data locality, minimizes privacy leakage, and reduces the risk of breaches.

The most widely implemented version of FL is the Federated Averaging (FedAvg) algorithm, which balances communication efficiency with model convergence. The algorithm works by averaging locally updated model parameters weighted by data volume at each client.

Algorithm 1: Federated Averaging (FedAvg)

```
Input: Initial global model $w_0$, number of clients N
For each round t = 1, 2, ..., T:
 Server:
 Select a subset of clients $S_t \subseteq N$
 Send global model $w_t$ to each client $i \in S_t$
 Each client i:
 Train on local dataset: $w_i \leftarrow w_t - \eta \nabla \ell_i(w_t)$
 Send $w_i$ to server
 Server:
 Aggregate: $w_{t+1} \leftarrow \Sigma_i \in S_t (n_i / \Sigma_j \in S_t n_j) * w_i$
```

Where:

w_t is the global model at time step t, η is the learning rate, $\ell_i(w_t)$ is the local loss function of client I, and n_i is the number of samples on client i.

Despite its privacy-conscious design, FL is vulnerable to a variety of adversarial attacks, especially in open, untrusted federated environments:

- **Model poisoning**: Malicious clients submit altered gradients to manipulate the global model.
- **Backdoor attacks**: A small number of poisoned inputs allow a client to mislead model predictions on specific triggers.
- **Gradient inversion**: Sensitive information is reconstructed from shared gradients (Zhu et al., 2019).

To combat these issues, several FL extensions have been proposed:

- **FedProx** (Li et al., 2020): Introduces proximal terms to account for system heterogeneity across clients.
- **Secure Aggregation** (Bonawitz et al., 2017): Protects intermediate updates from being observed by aggregators or eavesdroppers.
- **Differential Privacy-based FL**: Adds noise to gradients to obscure individual data contributions (Abadi et al., 2016).

However, these defenses remain largely classically bounded, meaning they rely on cryptographic assumptions that do not hold against quantum-capable adversaries. QFedSecure, by integrating post-quantum encryption and quantum key distribution, bridges this gap making it among the first federated protocols built to withstand adversaries armed with quantum decryption capabilities.

2.3. Trust Modeling in Distributed Systems

In federated environments, where clients span organizational, geographic, and policy boundaries, trust modeling becomes essential. Unlike traditional centralized systems with known actors and enforceable governance, federated systems must dynamically evaluate the reliability of each participant in real time.

Traditional approaches have relied on historical reputation scores or static access credentials, but these are insufficient in volatile or adversarial federations. Jøsang's Subjective Logic (2021) provides a mathematical framework for modeling uncertainty in trust relationships, particularly when evidence is partial or conflicting.

Equation 1: Subjective Trust Representation

Let: b: belief (evidence supporting trustworthiness), d: disbelief (evidence supporting untrustworthiness), and u: uncertainty (lack of sufficient information),

With the constraint:

$$b + d + u = 1$$

Then trust T is computed as:

$$T = b + \alpha u$$

Where $\alpha \in [0,1]$ reflects how much uncertainty contributes to the effective trust score. A lower α implies conservative trust assignment in the presence of limited evidence.

This model is particularly well-suited for federated learning scenarios characterized by frequent client churn, where participants may join and leave the federation unpredictably, and where communication is inherently asynchronous and intermittent. In such settings, direct verification of client behavior such as inspecting raw data is often infeasible due to privacy constraints and infrastructure heterogeneity.

Within the QAIFC framework, trust is not treated as a static attribute but rather as a dynamic, multi-factor function. It incorporates behavioral history, capturing deviations from expected model update patterns over time; cryptographic compliance, ensuring that participants adhere to verified quantum-safe protocols and produce valid attestations; and explainable anomaly detection, which leverages gradient-based or output-level explanations to identify suspicious or malicious contributions. This adaptive approach enables QAIFC to maintain trustworthiness in highly dynamic and partially observable federated environments.

These factors are processed by a trust scoring engine embedded within the Trust-Orchestration Layer (as described in Chapter 3). This engine plays a central role in regulating the behavior of participants in the federated system. It determines access to aggregation cycles, assigning privileges based on trustworthiness; it adjusts the weight of each client's contributions during model updates to mitigate the influence of unreliable nodes; and it governs participation in future training rounds, ensuring that only clients meeting predefined trust thresholds are allowed to continue collaborating.

Hybrid trust models combining subjective logic, machine learning classifiers, and cryptographic endorsements offer a robust path forward for building resilient federated systems that can scale across jurisdictions, use cases, and adversarial surfaces.

2.4. Explainable AI (XAI) in Security-Critical Systems

In highly sensitive and distributed systems such as federated learning across multiple cloud domains, the ability to explain and interpret AI decisions is not only a matter of user confidence, it is a vital security feature. As adversaries increasingly target AI pipelines through poisoning and stealthy manipulation, Explainable AI (XAI) emerges as a foundational component of trustworthy intelligence (Ghosh et al., 2022).

XAI enhances transparency by offering interpretations of model outputs or internals, enabling both users and automated systems to understand why a prediction was made. This is especially crucial when model updates are contributed by multiple clients, some of whom may be adversarial or compromised. Within QAIFC, XAI mechanisms are embedded into both client nodes and aggregation coordinators to analyze gradient behavior, flag anomalous updates, and adjust trust scores accordingly.

Popular XAI techniques can be categorized into two main types. Local interpretability methods focus on explaining individual predictions, making them particularly effective for debugging or analyzing specific gradient updates within a federated learning context. In contrast, global interpretability methods aim to provide insights into overall model behavior across entire datasets, making them valuable for assessing aggregate trust, identifying systemic biases, and supporting comprehensive system audits.

2.5. Popular XAI Techniques

- SHAP (SHapley Additive exPlanations): Based on cooperative game theory, SHAP assigns each feature an importance value for a particular prediction by simulating marginal contributions across feature combinations.
- LIME (Local Interpretable Model-Agnostic Explanations): Trains a lightweight surrogate model around a prediction by perturbing inputs and observing outputs.
- Grad-CAM (Gradient-weighted Class Activation Mapping): For convolutional neural networks, this technique visualizes which image regions most influenced the decision by backpropagating gradients.

Equation 2.1: SHAP Value for Feature i

$$\phi_i = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!\,(|F| - |S| - 1)!}{|F|!} [f(S \cup \{i\}) - f(S)]$$

Where:

- F: Set of all features, S: Subset of features excluding i, and $f(S)$: Model output with feature subset S.

This formulation ensures fair and additive attribution, making SHAP particularly well-suited for distributed FL scenarios where model transparency must be auditable and verifiable across domains.

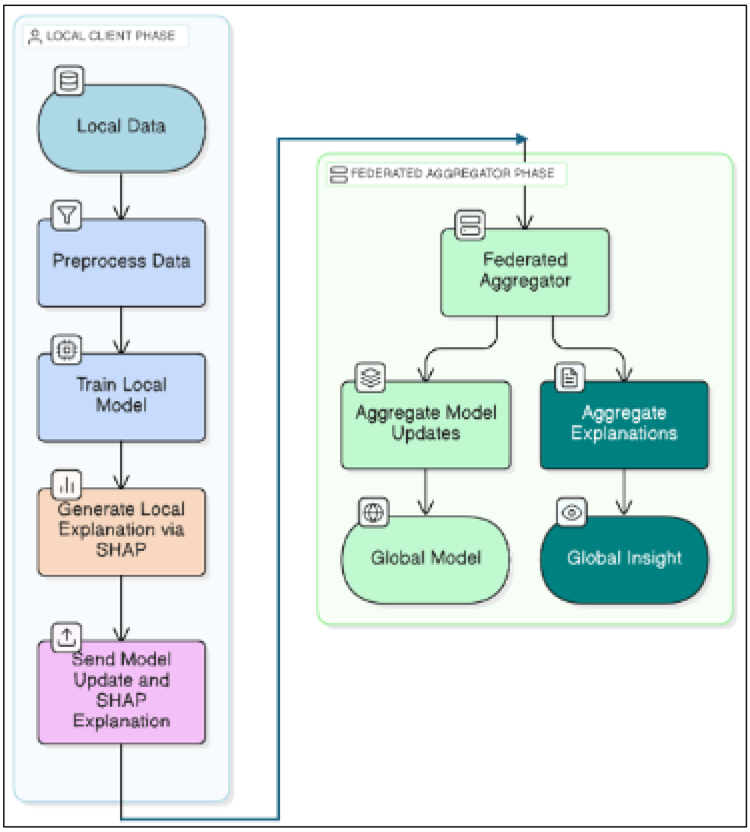

Figure 1 XAI Pipeline in Federated Learning

Integrating SHAP or LIME explanations into the federated learning update cycle allows the aggregation layer in QAIFC to perform several critical functions. It can identify and flag poisoned or misleading updates, ensuring that anomalous contributions do not compromise model integrity. It also quantifies the confidence of predictions, providing a measure of reliability that supports downstream decision-making. Additionally, it adjusts client trust scores based on the plausibility of explanations, aligning model behavior with expected patterns. Together, these capabilities bridge the gap between statistical accuracy and human interpretability, enhancing the safety, accountability, and regulatory compliance of federated learning systems, particularly in sensitive domains such as healthcare, law enforcement, and finance.

2.6. Cross-Domain Observability in Zero Trust Architecture

Traditional security models operate on implicit trust within network perimeters. However, federated learning across heterogeneous cloud and edge infrastructures undermines perimeter-based assumptions. The Zero Trust Architecture (ZTA) (Rose et al., 2020) replaces this with a “never trust, always verify” principle, requiring continuous authentication, verification, and monitoring of entities.

In QAIFC, observability is the operational pillar of ZTA, defined as the real-time collection, correlation, and analysis of telemetry data from federated nodes. Observability not only enables detection of malicious behaviors and drift but also provides the forensic backbone for post-event audits and dynamic trust recalibration.

2.6.1. Core Observability Dimensions

The core dimensions of observability in QAIFC encompass several critical aspects of system monitoring and accountability. Model updates are tracked in detail, capturing information about which clients submitted updates, when they were sent, and how they influenced the overall performance of the global model. Data provenance ensures that the origin and transformation history of inputs are verifiable, supporting auditability and trust in the training data. Access and identity logs authenticate how APIs are used and verify that participants comply with established policies. Finally, behavioral deviations are continuously monitored through analysis of usage patterns and anomaly scores, enabling early detection of abnormal or potentially malicious activity.

2.6.2. Components of Observability

The observability framework in QAIFC is supported by three key components, each serving a distinct role in maintaining system integrity and visibility. The Telemetry Agent is a lightweight module embedded within clients and servers that logs critical runtime data such as system calls, training statistics, gradient hashes, and anomaly scores. The Trust Broker functions as an intermediary authority that issues temporary trust tokens and validates each participant's behavioral history before permitting their updates to be aggregated. Meanwhile, the Policy Enforcer continuously assesses whether a node's current behavior adheres to established access control rules and assigned risk scores, ensuring that only compliant and trustworthy nodes can participate in the learning process.

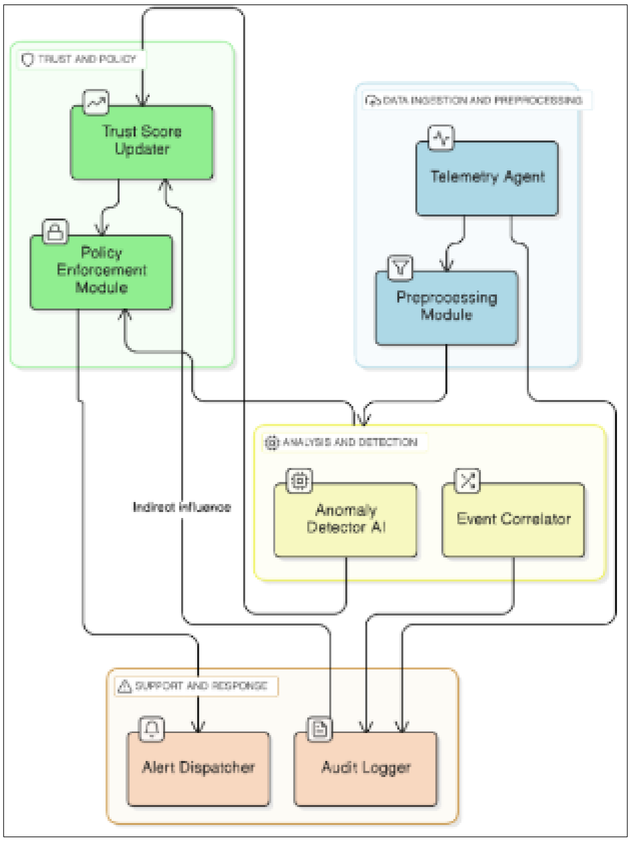

Figure 2 Observability Layers in Federated Cloud

This layered observability ensures that QAIFC not only reacts to malicious behaviors but anticipates and mitigates threats through continuous risk-aware operations.

Equation 2.2: Drift Detection in Federated Learning

$$D_{\{KL\}}(P\,||Q) = \sum_{i} P(i)\log\left(\frac{P(i)}{Q(i)}\right)$$

Where:

- P and Q: Probability distributions of predicted outputs over time windows, and $D_{\{KL\}}$: Kullback–Leibler divergence measuring statistical shift.

If the computed divergence exceeds a predefined threshold τ, the system initiates anomaly escalation protocols. This guards against silent model drift, targeted model poisoning, and data source spoofing.

Table 1 Technologies and Tools

Tool	Role in QAIFC
Prometheus + Grafana	Real-time telemetry collection and dashboard visualization
OpenTelemetry	Open-source instrumentation for distributed observability
Zeek	Network traffic analysis and policy anomaly detection

These observability platforms are container-native and integrate easily with Kubernetes-based FL deployments, making them ideal for scaling QAIFC across hybrid cloud environments.

2.7. Section-Specific Research Trends

Recent years have seen a convergence of multiple research domains relevant to QAIFC including federated learning, quantum cryptography, explainable AI, and trust modeling. The figure below illustrates the trajectory of scholarly publications (2018–2023) across the five key domains forming QAIFC's foundation.

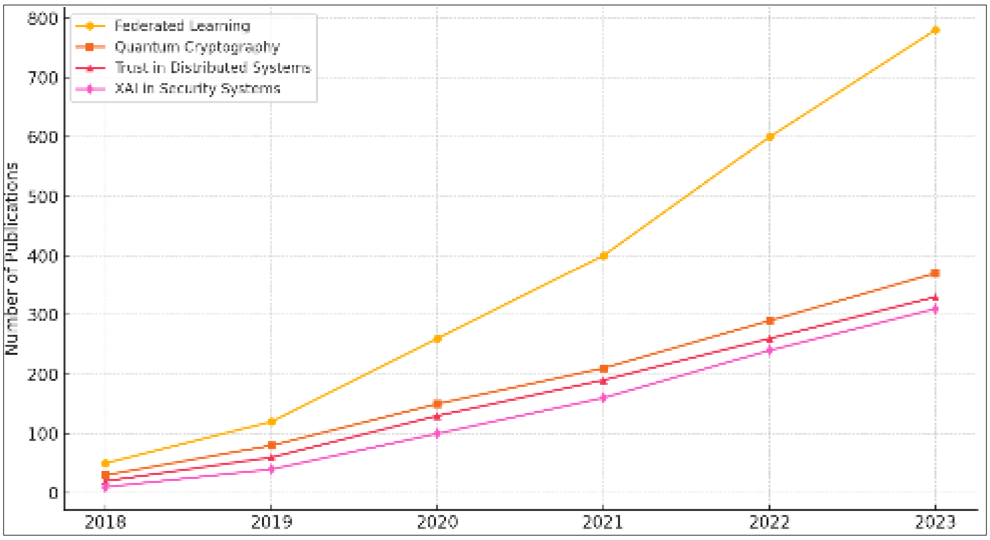

Figure 3 Publication Trajectory for Core QAIFC Domains (2018–2023)

Figure 3 Publication Trajectory for Core QAIFC Domains (2018–2023) illustrates the rapid growth and interdisciplinary convergence of the foundational domains supporting the QAIFC framework. Federated Learning publications surged from approximately 50 to over 880, largely fueled by applications in healthcare and edge AI. Trust Modeling also saw significant growth, driven by the need to manage identities and establish credibility in decentralized environments. The field of Quantum Cryptography expanded steadily, bolstered by progress in NIST's post-quantum cryptography (PQC) standardization efforts and the development of practical quantum key distribution (QKD) testbeds. Explainable AI (XAI) experienced rapid adoption, as ethical AI practices and legislation such as the EU AI Act demanded greater transparency and accountability in automated systems. Lastly, Observability emerged as a core concern with the mainstream adoption of Zero Trust Architecture and DevSecOps practices. Together, these trends underscore a growing academic and industry interest in robust, multi-layered security architectures, validating QAIFC's relevance as a forward-looking synthesis of federated intelligence, cryptographic resilience, explainability, and operational transparency.

2.8. Summary

Sections 2.2 through 2.6 have established the theoretical foundation for the Quantum-AI Federated Cloud (QAIFC) framework by synthesizing innovations from multiple interrelated domains. Federated Learning enables privacy-preserving orchestration of AI models by distributing training across client nodes without centralizing sensitive data. Trust Modeling facilitates dynamic, adaptive, and verifiable collaboration among heterogeneous participants, ensuring that contributions are both credible and accountable. Quantum Cryptography provides the cryptographic backbone necessary for future-proof communication and data protection, safeguarding against threats posed by quantum-capable adversaries. Explainable AI (XAI) adds a crucial layer of interpretability and auditability, allowing for transparent decision-making and detection of adversarial behavior within learning workflows. Meanwhile, Cross-Domain Observability brings the principles of Zero Trust into practice by enabling real-time, full-spectrum monitoring of system behavior across federated environments. Collectively, these pillars not only address existing vulnerabilities and limitations in federated architecture but also position QAIFC as a robust, scalable, and secure foundation for decentralized intelligence systems. The next chapter builds directly on these concepts by detailing the technical architecture and layered design of the QAIFC system.

3. Architectural Framework

3.1. Overview of QAIFC Architecture

The Quantum-AI Federated Cloud (QAIFC) represents a paradigm shift in distributed artificial intelligence specifically in how secure, trustworthy, and interpretable machine learning is orchestrated across heterogeneous, cross-domain environments. Traditional federated learning (FL) architectures have focused primarily on privacy preservation and performance optimization. However, they often assume semi-honest participants and rely on classical cryptographic primitives that are increasingly vulnerable to quantum computing attacks (Shor, 1994; Mosca, 2018).

QAIFC extends the capabilities of traditional federated systems by introducing a comprehensive multi-layered architecture that addresses emerging security and trust challenges in decentralized environments. It incorporates quantum-safe cryptographic primitives, such as lattice-based encryption and quantum key distribution, to ensure resilience against quantum-enabled threats. It also implements dynamic trust evaluation mechanisms, which leverage explainable AI and behavioral metrics to assess the credibility of each participant in real time. Additionally, QAIFC provides real-time observability across federated nodes, enabling continuous monitoring, auditability, and enforcement of Zero Trust Architecture principles throughout the system.

This design allows for decentralized and secure AI model training, even among mutually distrustful parties or across regulatory boundaries. It is particularly suited for security-critical domains such as digital healthcare, financial analytics, smart city control systems, and sovereign intelligence, where both data and infrastructure trustworthiness are paramount.

The QAIFC framework is modular and scalable, comprising five interdependent layers, each targeting specific systemic needs including:

- Application Layer – Provides interface points for users and services to access FL capabilities.
- Trust-Orchestration Layer – Manages node verification, trust scoring, and reputation-based access.
- Federated Learning and Coordination Layer – Executes the core machine learning workflows and coordination across clients.
- Quantum Secure Communication Layer – Ensures transport and exchange of encrypted model parameters using post-quantum and quantum-secure techniques.
- Cross-Domain Observability Protocol (CDOP) – Maintains visibility, anomaly detection, and telemetry across all nodes in the system.

Each layer supports a clear separation of concerns from interaction to execution, verification, and monitoring mirroring best practices in systems architecture, cybersecurity, and federated AI design (Zhou et al., 2021; Rose et al., 2020).

3.2. Layered Architecture Diagram

To visualize the QAIFC model, we introduce a layered reference architecture that illustrates the hierarchical structure and the interactions between these layers. The design aligns with cloud-native architectural standards and cybersecurity maturity models such as NIST SP 800-207 (Zero Trust Architecture), while also being extensible to domain-specific applications.

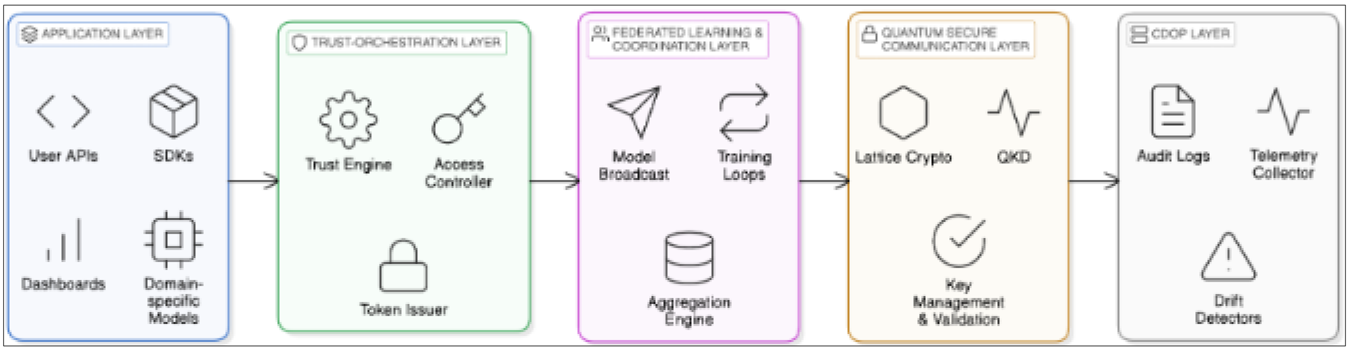

Figure 4 QAIFC Layered Architecture

Each block in the diagram represents an autonomous and modular layer, yet they interact through well-defined interfaces and security boundaries, ensuring layered defense and operational transparency.

3.2.1. Application Layer

This is the topmost layer through which end-users, data scientists, and domain-specific applications interact with the federated system. It includes:

- RESTful APIs, SDKs, and user dashboards
- Healthcare, finance, and industrial AI models
- Data preprocessing pipelines (on-device)

Applications developed in this layer do not directly handle sensitive data; instead, they send encrypted metadata or differential model inputs, relying on lower layers to enforce privacy and security (Bonawitz et al., 2019).

3.2.2. Trust-Orchestration Layer

The Trust-Orchestration Layer serves as the decision-making core for establishing, maintaining, and regulating trust within the QAIFC framework. It enforces dynamic access control and governs participation based on verifiable behavior, playing a pivotal role in securing federated collaboration. Drawing from principles of game theory and subjective logic (Jøsang, 2021), this layer continuously evaluates each client using adaptive trust scoring functions that incorporate both behavioral metrics such as gradient consistency and anomaly detection scores and cryptographic evidence, including attestations of secure protocol compliance.

The trust engine also issues short-lived trust tokens, which act as temporary credentials for accessing aggregation rounds. These tokens are generated and verified using zero-knowledge proof systems (e.g., zkSNARKs), ensuring that trust is granted based on proven behavior without disclosing sensitive details. Additionally, the layer functions as an intermediary between model performance and client reliability, modifying trust scores dynamically in response to observed outcomes.

By mediating participation through quantifiable trust, this layer is essential for defending against critical threats such as model poisoning, Sybil attacks, and freeloading, which can undermine federated learning in open and semi-trusted environments (Sharma et al., 2022). Its inclusion enables QAIFC to maintain system integrity, foster accountability, and support resilient, scalable AI collaboration across domains.

3.2.3. Federated Learning & Coordination Layer

The Federated Learning and Coordination Layer implements the core learning logic of the QAIFC framework, orchestrating collaborative model training across distributed, privacy-preserving clients. This includes the distribution of the global model from the central server to a dynamically selected subset of clients, local training on each client using private datasets, and the subsequent aggregation of updates using a trust-weighted mechanism to enhance security and resilience.

The standard Federated Averaging (FedAvg) protocol is extended in QAIFC to integrate trust scores (as detailed in Chapter 4), enabling the aggregation process to prioritize updates from more reliable participants. Additionally, encryption wrappers are applied to safeguard gradient updates in transit, and participation eligibility is enforced based on each client's current trust score and compliance status.

To support flexible and scalable training operations, this layer is designed to interface seamlessly with widely adopted federated learning frameworks such as TensorFlow Federated (TFF), PySyft, and FedML, allowing the QAIFC system to be deployed across diverse platforms and infrastructures while maintaining interoperability, security, and model performance.

3.2.4. Quantum Secure Communication Layer

As classical encryption becomes obsolete in the face of scalable quantum computing, this layer introduces quantum-resistant security primitives, such as:

- Lattice-based cryptography (e.g., Kyber, Dilithium, FrodoKEM)
- Quantum Key Distribution (QKD) simulations via Qiskit
- Key management servers (KMS) with verifiable entropy pools

Encrypted gradient updates are sealed end-to-end using keys generated via post-quantum schemes, ensuring forward secrecy and resistance against both present and future decryption attempts (Pirandola et al., 2020).

3.2.5. Cross-Domain Observability Protocol (CDOP) Layer

The Observability Layer is essential for maintaining the security, auditability, and real-time adaptability of the QAIFC framework. It provides comprehensive visibility into the behavior and health of the federated learning ecosystem by continuously collecting and analyzing diverse forms of operational data. This includes telemetry from clients, such as CPU utilization, local training durations, and cryptographic hashes of gradient updates, which helps detect inefficiencies or irregular behavior.

In parallel, the layer ingests anomaly detection outputs, including explainability-driven signals like SHAP-based flags, which are used to identify and respond to potential model poisoning or adversarial manipulation. It also monitors model drift signals, typically quantified using statistical measures like Kullback–Leibler (KL) divergence, to detect distribution shifts that may compromise model validity or performance.

By integrating these inputs, the Observability Layer enables dynamic policy enforcement, automated trust recalibration, and forensic auditability all critical functions for sustaining a Zero Trust federated infrastructure that is secure, transparent, and responsive.

Outputs from this layer feed into the Trust-Orchestration Layer, enabling adaptive trust recalibration and real-time revocation of malicious clients. It integrates tools like Prometheus, Grafana, OpenTelemetry, and Zeek for full-stack visibility (Makanju et al., 2020).

3.3. Application Layer

3.3.1. Function

The Application Layer serves as the interface between end-users, system administrators, and the broader federated infrastructure. It is the only layer directly exposed to users and is designed to abstract the complexity of secure federated learning operations, offering a seamless user experience across multiple platforms. This layer enables domain-specific applications to submit model requests, visualize results, and interact with trained models without needing access to underlying data or cryptographic protocols (Kairouz et al., 2021).

Application contexts for QAIFC are diverse and span several high-impact domains. These include collaborative AI modeling for cross-institutional research, where academic or industrial entities contribute to joint model development without exposing proprietary datasets. In healthcare diagnostics, multiple hospitals or clinics can participate in federated learning to improve diagnostic models while strictly maintaining the privacy of patient data in accordance with regulations such as HIPAA. In the financial sector, QAIFC supports fraud detection and risk analysis by enabling banks and financial institutions to collaboratively train models on transaction patterns while ensuring confidentiality. Additionally, in IoT and edge computing environments, QAIFC facilitates decentralized coordination among smart sensors, allowing them to contribute data and consume real-time AI inferences without centralized bottlenecks.

To support these varied use cases, the Application Layer plays a crucial role in ensuring usability, security, and regulatory compliance. It achieves this by interfacing securely with the Trust-Orchestration Layer through well-defined API calls, access tokens, and telemetry submission protocols, thereby maintaining seamless and trusted interactions between user-facing applications and the secure federated backend.

3.3.2. Responsibilities

The Application Layer of the QAIFC framework carries several critical responsibilities that ensure smooth and secure interaction between end-users, federated clients, and the underlying orchestration mechanisms. It handles User Input/Output, enabling the system to capture user-defined training goals, model queries, and feedback, while returning interpretable outputs such as predictions or SHAP-based visual explanations. Through its Data Preprocessing Pipelines, the layer transforms raw inputs such as CSV files, sensor logs, or image arrays into a structured format suitable for local federated training. Importantly, this process ensures that raw data remains confined to the client device, preserving privacy and compliance.

The layer also facilitates Client-Server Interaction, where each client securely authenticates with the Trust-Orchestration Layer, receives the current global model, executes local training, and transmits encrypted updates back to the server. Lastly, the Application Layer supports Integration Hooks that allow external services such as compliance checkers, visualization dashboards, and federated benchmarking tools—to seamlessly interface with QAIFC workflows. These responsibilities ensure not only secure and compliant operation but also enhance the usability and adaptability of the entire federated learning ecosystem.

Table 2 Tools & Technologies

Tool	**Functionality**
TensorFlow Federated (TFF)	Enables simulation and deployment of client-side federated learning.
Web and Mobile SDKs	Build native client applications with built-in FL compatibility.
Streamlit, Dash, Grafana	Provide real-time dashboards for model monitoring, visualization, and feedback.
RESTful APIs / GraphQL	Facilitate access to model status, telemetry data, and trust scores via standard protocols.
Docker / Kubernetes	Host scalable, containerized client apps across edge environments.

By combining these technologies, the application layer can support **zero-trust, multi-party learning systems**, making it suitable for both real-time analytics and batch training in sensitive sectors.

3.4. Trust-Orchestration Layer

3.4.1. Function

The Trust-Orchestration Layer is the core decision-making module of the QAIFC stack. It is responsible for evaluating, scoring, and regulating every participant in the federated ecosystem based on dynamic behavioral, cryptographic, and statistical metrics. Unlike static access control lists or identity verification alone, this layer actively adjusts client trust scores in real time, allowing the system to tolerate transient failures and reject malicious actors (Sharma et al., 2022; Jøsang, 2021).

This dynamic trust model enables risk-aware access control within federated learning by supporting several key functions that enhance both security and operational integrity. It allows for the real-time exclusion of suspicious nodes from the aggregation process, thereby preventing potentially harmful updates from corrupting the global model. It also facilitates trust-weighted gradient contributions, where the influence of each client’s update is proportionate to its calculated trust score, ensuring that reliable nodes have greater impact.

In addition, the model supports anomaly-aware training incentives and penalties, dynamically rewarding honest participation and penalizing detected adversarial behavior. Trust decisions are further strengthened through decentralized attestation mechanisms, combining cryptographic signatures with insights from explainable AI to verify the legitimacy and transparency of each participant’s actions.

This layer is critical to ensuring that federated AI systems are not only auditable but also resilient, particularly in environments where adversaries may be present or computational resources are limited. By embedding adaptive, trust-aware logic into the core learning process, QAIFC establishes a secure and accountable foundation for decentralized intelligence.

3.4.2. Key Components

The Trust-Orchestration Layer in QAIFC is built on several interdependent components that work together to evaluate and enforce trustworthy participation in the federated learning process.

- **Trust Score Calculator**: Computes scores based on multiple weighted metrics, including behavior, model quality, cryptographic integrity, and peer endorsements.
- **Token-Based Access Verifier**: Issues and verifies trust tokens that regulate participation eligibility, expiration, and revocation.
- **Reputation Database**: Stores long-term and session-based behavior logs, performance statistics, and incident reports for each participant.
- **Explainable AI Verifier**: Evaluates model updates using SHAP or LIME explanations to determine whether an update's behavior is consistent with its input data and expected feature attribution patterns.
- Formula 3.1: Dynamic Trust Score Function

$$T_{ij}(t) = \frac{\sum_{k=1}^{n} w_k S_k(i,j,t)}{\sum_{k=1}^{n} w_k}$$

Where: $T_{ij}(t)$: Trust score assigned by node j to node i at time t, $S_k(i,j,t)$: The k-th trust metric (e.g., anomaly rate, historical accuracy, cryptographic validation), w_k: Weight assigned to metric k based on its reliability and sensitivity.

This multi-dimensional trust function ensures a balanced and tunable scoring system that can reflect diverse operational realities, such as higher trust for historically stable nodes or penalization for sudden behavioral shifts.

3.4.3. Trust Metrics May Include

Trust metrics in QAIFC are designed to provide a nuanced, real-time assessment of each participant's reliability, leveraging a combination of behavioral analytics, cryptographic verification, and peer validation. One key metric is gradient deviation, which measures how significantly a client's submitted gradient diverges from the statistical average of its cohort. Large deviations may indicate model poisoning or data inconsistency, triggering further scrutiny. Historical performance is another critical factor, capturing each client's contribution quality and consistency over time, and rewarding nodes that demonstrate stable, accurate behavior.

Cryptographic compliance ensures that participants adhere to the system's security standards by verifying the use of quantum-safe signatures, post-quantum encryption schemes, and enforcement of key expiration policies. This ensures that even in a future quantum-enabled threat environment, the system remains secure.

Additionally, peer endorsements allow clients to vouch for each other through mechanisms such as cryptographic referrals or blockchain-based attestations, adding a decentralized layer of trust validation. These endorsements can influence initial trust scores or serve as secondary validators when anomalies are detected.

All trust metrics are continuously recalibrated using real-time telemetry and interpretability data collected through the Cross-Domain Observability Protocol (CDOP). This enables the trust engine to adapt dynamically to changing client behavior and evolving threat landscapes, reinforcing QAIFC's resilience and accountability.

3.4.4. Applications

The trust mechanisms implemented within QAIFC support a wide range of critical applications that enhance both the security and operational resilience of federated learning environments. One such application is the dynamic inclusion or exclusion of nodes from the aggregation process, where participants are admitted or barred based on real-time trust thresholds and anomaly detection scores. This ensures that only trustworthy clients influence the global model, significantly reducing the system's vulnerability to adversarial behavior.

Another key application is trust-weighted model aggregation, in which each client's contribution is scaled according to its computed trust score. This approach improves the robustness of the global model by minimizing the influence of low-trust or suspicious updates without requiring their outright exclusion.

QAIFC also supports real-time forensic logging, enabling transparent audits of trust-based decisions. This feature provides accountability and traceability, particularly in scenarios involving disputes over false positives or negatives in trust evaluation. Logs can be reviewed to understand why a particular client was penalized or excluded, supporting governance and compliance.

Finally, the trust engine enables automated enforcement of security policies, such as blocking any client whose trust score falls below 0.3 for three consecutive rounds. These programmable policies allow administrators to implement and adapt risk controls in real time, maintaining system integrity across changing conditions and threat landscapes.

3.5. Federated Learning & Coordination Layer

3.5.1. Function

The Federated Learning and Coordination Layer is the operational core of QAIFC architecture. It is responsible for orchestrating decentralized training across geographically dispersed, privacy-sensitive nodes while ensuring consistency, integrity, and convergence of the global machine learning model. Unlike traditional centralized ML pipelines, this layer supports a distributed computation paradigm, where each client trains a local model using private data and contributes encrypted and trust-weighted updates to the shared global model (McMahan et al., 2017).

Key differentiators of this layer in QAIFC include:

- Trust-aware aggregation, where model updates are scaled based on a client's current trust score (computed in the Trust-Orchestration Layer).
- Secure model parameter exchange using post-quantum and quantum-secure communication protocols (see Section 3.6).
- Integration with anomaly detection feedback loops, which adapt aggregation weightings based on real-time model explainability metrics.

3.5.2. Workflow Steps

The learning process in this layer proceeds in the following structured steps, repeated over multiple communication rounds:

Model Initialization & Broadcast

The global model W_t is initialized or updated and then broadcasted from a central aggregator or decentralized peer coordinator to selected clients S_t.

Local Training with Private Data

- Each selected client performs training on its private dataset D_i without uploading the data itself.
- The local optimization follows stochastic gradient descent (SGD) or its variants.

Trust-Weighted Aggregation

- Clients return encrypted gradient updates W_i along with their trust score T_i.
- The server aggregates updates proportionally to their trust values using Equation 3.1 from the Trust-Orchestration Layer.

Feedback to Orchestration Layer

- Model behavior (e.g., convergence speed, SHAP consistency, anomaly scores) is sent to the Trust-Orchestration Layer for recalibration of trust values.
- This design allows QAIFC to dynamically adapt to changing node behavior, detect faulty or malicious participants early, and optimize model performance while preserving user privacy and security (Kairouz et al., 2021).

Algorithm 3.1: Trust-Weighted Aggregation

```
For each communication round t:
  For each client i in selected set $S_t$:
  Send model $W_t$ to client i
  Client i performs training: $W_i \leftarrow W_t - \eta\nabla\ell_i(W_t)$
  Client i returns encrypted $W_i$ and trust score $T_i$
  Server aggregates:
  $W_{t+1} \leftarrow \Sigma (T_i * W_i) / \Sigma T_i$
```

Where:

- W_t: Current global model, η: Learning rate, $\nabla\ell_i(W_t)$: Gradient of the local loss function at client iii, and T_i: Trust score from the Trust-Orchestration Layer

This algorithm increases robustness against model poisoning and Sybil attacks by reducing the aggregation influence of low-trust clients while encouraging well-behaved participants.

Table 3 Technologies

Tool	**Role**
TensorFlow Federated (TFF)	Simulation and prototyping of federated learning workflows
FedML	Scalable deployment for real-world edge/cloud FL coordination
PySyft	Secure aggregation with encrypted tensors and privacy guarantees
MLflow	Logging of model versions, hyperparameters, and metrics across rounds
KubeFL / Flower	Kubernetes-based FL orchestration across multi-cloud clusters

These tools make it possible to deploy QAIFC at scale, with observability hooks and real-time update routing across clients and aggregation servers.

3.6. Quantum Secure Communication Layer

3.6.1. Function

The Quantum Secure Communication Layer plays a critical role in safeguarding the integrity and confidentiality of all data transmissions within the QAIFC framework. As federated learning systems increasingly operate over open and potentially untrusted infrastructures such as public cloud platforms, 5G edge nodes, and multi-domain networks the reliance on post-quantum and quantum-resilient cryptographic solutions becomes imperative.

This layer ensures that gradient updates, trust tokens, and telemetry logs are protected using quantum-safe cryptographic primitives, including lattice-based encryption and ring-LWE constructions. To secure communication sessions, it supports the establishment of symmetric session keys through either post-quantum key exchange protocols (e.g., Kyber, FrodoKEM) or Quantum Key Distribution (QKD) channels when available, providing resilience even in the presence of quantum-capable adversaries.

To further reinforce integrity and identity verification, the layer integrates zero-knowledge signatures and tamper-evident logging mechanisms, ensuring that data in transit is both authenticated and auditable without compromising privacy. All communication protocols within this layer are designed to align with emerging NIST and ETSI security standards, positioning QAIFC for seamless transition into a post-quantum cryptographic landscape (Alkim et al., 2016; Pirandola et al., 2020). By embedding these security guarantees at the communication level, QAIFC ensures that federated learning remains trustworthy and secure, even under the most advanced threat models.

3.6.2. Features

The Quantum Secure Communication Layer in QAIFC integrates multiple cryptographic techniques to ensure robust protection of federated learning data, even in the face of future quantum threats.

- **Post-Quantum Key Exchange (PQKE)** utilizes lattice-based encryption schemes such as Kyber, FrodoKEM, and Dilithium to establish secure communication channels. These schemes are designed to withstand attacks from quantum computers, providing forward secrecy and ensuring that previously exchanged data remains secure even if future keys are compromised.
- **Quantum Key Distribution (QKD)**, specifically through the implementation of the BB84 protocol, allows the generation of provably secure symmetric keys based on the principles of quantum entanglement and measurement. This approach inherently protects against eavesdropping, as any attempt to intercept the quantum transmission results in a detectable disturbance due to quantum-state collapse.
- To authenticate all transmitted data, QAIFC also employs End-to-End Message Authentication Codes (MACs). These MACs verify the integrity and authenticity of each communication packet, effectively preventing man-in-the-middle and replay attacks. The MACs are built using quantum-safe hashing algorithms, such as SHA-3 and SPHINCS+, which are resistant to both classical and quantum cryptanalytic techniques.

Together, these technologies form a multilayered security model that ensures confidentiality, integrity, and forward secrecy in federated AI communications, aligning with post-quantum cryptographic standards.

3.6.3. Script Example: BB84 QKD Simulation Using Qiskit

```
from qiskit import QuantumCircuit, execute, Aer

qc = QuantumCircuit(1, 1)
qc.h(0) # Step 1: Create a superposition state
qc.measure(0, 0) # Step 2: Measure in computational basis

backend = Aer.get_backend('qasm_simulator')
result = execute(qc, backend, shots=1024).result()
counts = result.get_counts()
print("Simulated BB84 bit counts:", counts)
```

This example simulates one step in the BB84 QKD protocol, demonstrating key generation and measurement. In production systems, these keys would be validated via error checking and privacy amplification before use in federated encryption routines.

Table 4 Standards & Protocols

Standard	**Description**
NIST PQC Suite	Cryptographic algorithms selected for post-quantum resistance (Kyber, Dilithium, Falcon)
ETSI GS QKD 004	Protocols and architectural guidelines for deploying QKD networks
IETF CFRG	Quantum-safe hash functions and secure key derivation mechanisms

These standards ensure interoperability, security assurance, and compliance for deployments in critical sectors such as finance, healthcare, and national security.

3.6.4. Use Case Integration

- In a multi-hospital federation, QKD channels could be established between data centers to prevent MITM attacks on patient model exchanges.
- In a national security application, session keys derived from Kyber could protect telemetry logs and trust certificates exchanged between distributed intelligence nodes.

3.7. Cross-Domain Observability Protocol (CDOP)

3.7.1. Function

The Cross-Domain Observability Protocol (CDOP) serves as the real-time sensory and monitoring infrastructure for QAIFC, ensuring operational transparency and traceability across all federated components. Traditional federated learning architectures lack fine-grained visibility into data provenance, update quality, and policy adherence, making them vulnerable to stealthy adversarial activity and unexplainable performance degradation.

CDOP addresses this gap by enabling continuous:

- Telemetry collection from client and server nodes,
- Anomaly classification using explainable AI,
- Drift detection to identify model corruption or conceptual shifts,
- Security auditing and compliance reporting across jurisdictional boundaries.

Inspired by observability principles in Zero Trust Architecture (ZTA) and modern DevSecOps systems (Rose et al., 2020), CDOP transforms QAIFC into a self-aware, self-defending learning ecosystem.

Table 5 Core Modules

Module	**Description**
Telemetry Capture Agents	Embedded agents collect fine-grained logs (e.g., training time, loss values, API usage) from each federated node.
Anomaly Detection Engine	Leverages XAI methods (e.g., SHAP, Isolation Forest) to score each update and identify malicious or inconsistent behaviors.
Audit Log Replicator	Aggregates and distributes audit trails for tamper-proof storage across domains using ELK stack or blockchain-based ledgers.
Drift Tracker	Continuously compares prediction distributions against historical norms using statistical divergence metrics.

Formula 3.2: Model Drift Detection Using KL Divergence

Model drift is a key indicator of untrustworthy contributions or emerging data shifts. CDOP applies Kullback-Leibler Divergence (KL Divergence) to compare predicted output distributions over time.

$$D_{KL}(P\,||Q) = \sum_{i} P(i)\log\left(\frac{P(i)}{Q(i)}\right)$$

Where:

- $P(i)$: Expected or historical output distribution (reference), $Q(i)$: New/current output distribution (observation), and D_{KL}: Divergence measure (non-symmetric).

When $D_{KL} > \tau$, where τ is a predefined threshold (typically 0.1–0.3), CDOP triggers:

- Trust reassessment in the Trust-Orchestration Layer,
- Quarantine or rate-limiting of the suspected client,
- Logging of the drift event into the distributed audit ledger.

This detection technique helps in mitigating silent model corruption, data poisoning, and domain shifts (Lu et al., 2019).

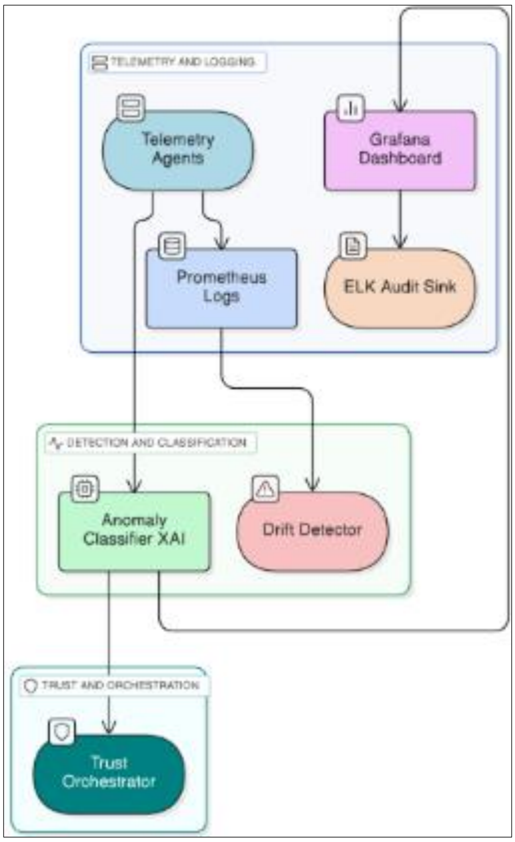

Figure 5 CDOP Data Flow

This architecture allows QAIFC to observe, react, and adapt to evolving threats and performance deviations across any federated domain, cloud provider, or organizational boundary.

Table 6 Integration Tools and Platforms

Tool/Framework	**Role**
OpenTelemetry	Cross-platform telemetry instrumentation and metric streaming.
Grafana	Visualization dashboard for trust scores, drift alerts, and update frequency.
Zeek IDS	Passive network monitoring tool for detecting suspicious traffic or protocol violations.
ELK Stack (Elasticsearch, Logstash, Kibana)	Distributed log collection, search, and real-time audit visualization.
Prometheus	Time-series database and alerting engine for metrics gathered from all layers.

Together, these tools form a robust observability and alerting framework for federated AI, enabling compliance, rapid remediation, and trust transparency.

3.7.2. CDOP Use Case Example

In a QAIFC deployment across three healthcare systems:

- Telemetry agents monitor training behavior of hospital A's edge node.
- Anomaly classifier flags inconsistent gradient directions compared to the rest of the federation.
- Drift tracker confirms a sharp rise in KL divergence.
- CDOP initiates trust downgrading, logs the incident, and temporarily removes hospital A's node from aggregation until a remediation protocol is executed.

This ensures real-time trust regulation and privacy-preserving accountability without centralizing sensitive patient data.

3.8. Inter-Layer Communication and Feedback Loops

The QAIFC architecture is not simply a stack of isolated functionalities; it is a deeply interconnected system where each layer shares metrics, triggers, and policy feedback with its neighbors. This multi-layer feedback loop creates a resilient, context-aware AI platform that can enforce security, optimize performance, and adapt to real-world variability.

3.8.1. Examples of Inter-Layer Interactions

The QAIFC framework is designed with tightly integrated inter-layer communication to ensure that security, trust, and observability are enforced coherently across all functional domains. For instance, trust scores computed in the Trust-Orchestration Layer are directly informed by inputs from other layers specifically, real-time anomaly scores generated by the Cross-Domain Observability Protocol (CDOP) and cryptographic proof compliance verified by the Quantum Secure Communication Layer. This allows the trust engine to reflect both behavioral irregularities and cryptographic adherence in its scoring.

In the Federated Learning Layer, model aggregation decisions are governed by these trust scores. Only updates from clients that meet the current trust threshold are included in the global model, and even among those, the gradient contributions are scaled according to trust-weighted factors as described in Section 3.4. This mitigates the influence of potentially unreliable or marginally trusted participants.

Meanwhile, audit logs and telemetry data collected by CDOP are stored in tamper-evident formats and can be queried through the Application Layer by system administrators via secure dashboards or APIs. This provides visibility into system health and supports compliance reporting and incident forensics.

Further reinforcing system integrity, gradient updates are only accepted and processed if they meet multiple conditions: they must be signed with valid post-quantum certificates, the associated trust score must exceed a minimum threshold (e.g., 0.6), and there must be no recent drift or anomaly flags tied to the client's activity.

This holistic, cross-layer design ensures that attacks, distributional drift, or component failures are rapidly detected, isolated, and addressed by adjacent system layers. It also enables federation-wide transparency and resilience, supporting compliance with emerging AI governance standards and data sovereignty regulations (Kumar et al., 2021).

3.9. Summary

QAIFC architecture introduces a modular, layered approach to securing federated learning in untrusted and quantum-vulnerable environments. Each layer from user-facing applications to quantum-secure communication and real-time observability contributes to a coherent and robust federated ecosystem. At the core lies a dynamic trust function, allowing adaptive participation and cryptographically verified collaboration across administrative domains. This layered model sets the foundation for the protocol and threat modeling introduced in Chapter 4 and 5.

4. Proposed Protocol: QFedSecure

4.1. Introduction

In federated learning environments involving untrusted, heterogeneous participants particularly those spanning multiple cloud domains and geopolitical boundaries ensuring security, accountability, and trust becomes paramount. The rise of quantum computing further threatens classical encryption, compelling the need for protocols that are not only distributed and resilient but also quantum secure.

QFedSecure addresses the core challenges of secure, decentralized AI collaboration by introducing a composable federated learning protocol that integrates protection, trust, and interpretability across five critical dimensions. First, it incorporates Quantum Key Distribution (QKD) and post-quantum encryption to secure communications against quantum-enabled adversaries, ensuring forward secrecy and long-term confidentiality (Pirandola et al., 2020). Second, it leverages lattice-based cryptographic primitives, such as Kyber and Dilithium, to encrypt model parameters and verify participant identities with resistance to both classical and quantum attacks (Alkim et al., 2016).

Third, the protocol embeds AI-driven anomaly detection, enabling the system to identify poisoned or otherwise suspicious updates before they influence the global model. Fourth, it employs dynamic trust scoring, assigning weights to client contributions based on behavioral integrity, cryptographic compliance, and explainability metrics. Finally, feedback loops are implemented to reinforce trustworthy behavior over time, allowing the system to adaptively reward reliable nodes and penalize those exhibiting malicious or erratic behavior.

QFedSecure is modular and pluggable, designed to integrate seamlessly with federated learning orchestration frameworks such as TensorFlow Federated, PySyft, and FedML. Its alignment with Zero Trust principles, commitment to cryptographic assurance, and support for explainable intelligence make it particularly well-suited for high-assurance environments, including healthcare, financial services, national defense, and critical infrastructure sectors where trust, security, and transparency are non-negotiable.

4.2. QFedSecure Protocol Design and Specification

The QFedSecure protocol is executed in five sequential phases, each contributing to the end-to-end lifecycle of secure and verifiable federated learning:

- Quantum Key Initialization – Clients and coordinators negotiate session keys using QKD or post-quantum key exchange to ensure secure communication.
- Secure Model Distribution – Clients receive encrypted model parameters after passing access and trust verification checks.
- Anomaly Detection Pipeline – Local updates are evaluated using local explainability techniques (e.g., SHAP) to detect inconsistencies or adversarial intent.
- Gradient Encryption and Trust Scoring – Gradients are encrypted with quantum-safe primitives and assigned a dynamic trust score based on observed behavior and past reputation.
- Secure Aggregation and Trust Feedback – Aggregators perform trust-weighted update aggregation and send scores back to the Trust-Orchestration Layer.

This compositional pipeline enables real-time risk-aware learning, mitigating both insider threats and long-range adversarial attacks without centralizing any sensitive data.

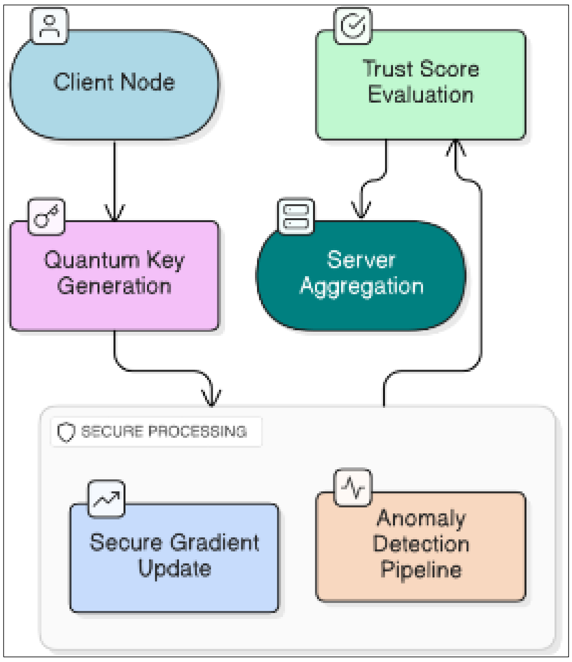

Figure 6 QFedSecure Protocol Flow

Each component of this pipeline is cryptographically protected, trust-evaluated, and observable via telemetry integrations from the CDOP layer.

4.3. Quantum Key Exchange for Federated Learning

To protect model parameters and trust credentials in transit, QFedSecure integrates Quantum Key Distribution (QKD) into its initial communication handshake. The selected protocol for simulation and early implementation is BB84, the first quantum cryptographic protocol proposed by Bennett and Brassard in 1984.

BB84 exploits the no-cloning theorem and quantum measurement collapse to enable two parties (typically a client and a server) to establish a symmetric key that is immune to interception even by quantum adversaries (Scarani et al., 2009). Unlike RSA or ECC, BB84 does not rely on hard math problems, making it probably secure under quantum and classical threats.

Algorithm 4.1: BB84 QKD Handshake (Simplified)

```
1. Alice prepares qubits in random bases (X or Z)
2. Bob measures qubits in random bases
3. Alice and Bob communicate their bases over a public channel
4. Discard mismatched bases to extract shared key
5. Apply privacy amplification to remove leaked bits
```

This key is then used to encrypt gradients using symmetric lattice-encryption schemes (e.g., Kyber) or AES-256 (fallback) depending on quantum hardware availability.

4.3.1. Script: Simulating Quantum Key Generation Using Qiskit

The following Python script demonstrates a simplified QKD key generation using IBM's Qiskit simulator, ideal for educational and prototyping purposes:

```
from qiskit import QuantumCircuit, Aer, execute

qc = QuantumCircuit(1, 1)
qc.h(0) # Create superposition
qc.measure(0, 0) # Measure in Z basis

backend = Aer.get_backend('qasm_simulator')
result = execute(qc, backend, shots=1024).result()
print("Key bits:", result.get_counts())
```

The outcome shows a probabilistic distribution of qubit measurement results, which represent one half of the key pair. A second party would run a similar protocol and compare measurement bases to generate the final key.

4.3.2. Security Implications

BB84 offers forward secrecy keys are freshly generated per session and cannot be retroactively decrypted even if an attacker captures the encrypted gradients. Moreover, BB84 and its extensions (e.g., decoy state BB84) can detect eavesdropping by measuring the quantum bit error rate (QBER).

QFedSecure can also fall back to post-quantum key exchange protocols such as Kyber or FrodoKEM, which are efficient and NIST-finalist candidates for deployment in classical environments (NIST PQC, 2023).

4.4. Secure Gradient Exchange Using Lattice-Based Cryptography

In the post-quantum era, classical encryption schemes such as RSA and ECC become vulnerable to decryption using algorithms like Shor’s, which run in polynomial time on quantum hardware (Shor, 1994). To prevent model inversion, gradient leakage, and unauthorized update analysis, QFedSecure leverages lattice-based cryptography specifically schemes based on the Learning With Errors (LWE) problem.

LWE is considered quantum-resistant and forms the foundation of cryptographic primitives such as Kyber, FrodoKEM, and Dilithium, which are all NIST-approved for post-quantum standardization (Alkim et al., 2016; NIST PQC, 2023).

Formula 4.1: LWE-Based Gradient Encryption

Let:

A: Public matrix, s: Secret vector (private key), e: Small noise vector sampled from a discrete Gaussian distribution, q: A large prime modulus.

The encryption of a gradient vector s proceeds as follows:

$$b = A\,s + e\,mod\,q$$

To decrypt:

$$smod\ qs \ \approx b - A$$

Since e is small and known in distribution, the decrypted signal can recover s with high probability. This noise also hides the structure of s, ensuring semantic security under quantum attacks.

4.4.1. Use in QFedSecure

In the QFedSecure protocol, lattice-based encryption is employed to ensure secure and privacy-preserving communication of model updates between clients and the aggregator, even in the presence of quantum-capable adversaries. The process begins with each client encrypting its local model updates either gradients or weights using a public matrix AAA and their own private key sss. This encryption is based on the Learning With Errors (LWE) problem, which underpins the security of modern lattice-based schemes such as Kyber and FrodoKEM.

Once encrypted, these updates are transmitted via the Quantum Secure Communication Layer, which guarantees end-to-end confidentiality using post-quantum key exchange or quantum key distribution (QKD). Importantly, the aggregator does not decrypt these individual updates. Instead, it performs homomorphic aggregation such as computing a sum or mean directly on the ciphertexts. This allows the global model to be updated securely without ever exposing the individual contributions of clients.

By enabling homomorphic operations over encrypted data, this mechanism preserves both data privacy and model integrity, while remaining resistant to quantum attacks. It also ensures that no single point in the system, including the central aggregator, ever has access to raw or decrypted client data, making QFedSecure compliant with the strongest privacy and cryptographic guarantees.

Table 7 Cryptographic Tools & Libraries

Tool	Function
PQC-CRYSTALS-Kyber	Official NIST finalist for post-quantum KEM. Used for encrypting gradients and keys.
OpenFHE	Fully homomorphic encryption framework for secure computation over encrypted data.
PALISADE	Lattice-based homomorphic encryption library for FL-compatible secure aggregation.
Liboqs	Open Quantum Safe implementation for integrating multiple PQC algorithms.

This cryptographic workflow ensures forward secrecy, model confidentiality, and secure gradient exchange without exposing the local data, gradient patterns, or update behaviors of individual clients.

4.5. AI-Driven Anomaly Detection Pipeline

Even with robust cryptographic protection in place, federated learning systems remain vulnerable to a range of data-driven attacks that exploit the statistical nature of model updates rather than the transmission channels themselves. Among the most critical threats are model poisoning, where adversaries submit manipulated gradients to degrade the overall model accuracy; backdoor insertion, which involves injecting specific patterns that trigger malicious misclassifications; gradient inversion, where attackers attempt to reconstruct private training data from shared gradients; and Sybil attacks, where a single entity impersonates multiple clients to disproportionately influence the model.

To counter these threats, QFedSecure incorporates an embedded AI-driven anomaly detection engine within its Trust-Orchestration Layer. This engine evaluates each client's model update by analyzing behavioral patterns, measuring statistical deviations from expected gradient distributions, and assessing consistency over time. Clients whose updates exhibit abnormal behavior such as extreme gradient magnitudes, divergent loss trajectories, or uncharacteristic feature attributions are assigned to elevated risk scores. These scores directly inform trust calculations, enabling the system to penalize or exclude malicious actors in real time. This integrated approach provides a second line of defense beyond cryptography, reinforcing the overall resilience and reliability of the QAIFC framework.

Table 8 Pipeline Components

Component	**Role**
SHAP Explainability	Explains local gradient updates by attributing them to feature contributions.
Isolation Forests	Detects outliers in gradient magnitude, direction, or model loss.
KL-Divergence Drift Detection	Measures temporal shift in model output distribution to detect concept drift.

Equation 4.2: KL-Divergence for Drift Detection

$$D_{KL}(P\,||Q) = \sum_{i} P(i) \log\left(\frac{P(i)}{Q(i)}\right)$$

Where:

- $P(i)$: Expected distribution of model outputs (from prior rounds), $Q(i)$: Current observed distribution,
- D_{KL}: Measures divergence; large values indicate unexpected shifts.

If $D_{KL}(P\,||Q) > \tau$ (a threshold like 0.3), it is flagged as suspicious and can:

Lower the client's trust score, Delay or isolate its update from aggregation, and/or Trigger administrative audit or visualization via the CDOP layer.

4.5.1. Tools for Implementation

- SHAP (SHapley Additive exPlanations): For local gradient attribution.
- PyCaret / Scikit-Learn: For deploying unsupervised anomaly detectors.
- Grafana + Prometheus: For real-time metric visualization and anomaly alerts.

This enables explainable, autonomous vetting of client updates and feeds results directly to the trust engine.

4.6. Trust Score Evaluation and Secure Aggregation

In QFedSecure, each client is assigned a dynamic trust score that continuously evolves based on observed behavior, contribution reliability, and adherence to cryptographic protocols. These trust scores serve as a central mechanism for enforcing accountability and security within the federated system. During model aggregation, trust scores are used to weight each client's contribution, ensuring that updates from more reliable participants have greater influence on the global model. They also govern access privileges, determining which clients are eligible to receive updated models or participate in subsequent communication rounds.

Furthermore, trust scores are persistently stored and visualized within the Cross-Domain Observability Protocol (CDOP) telemetry stack. This enables real-time monitoring, historical analysis, and auditability of trust dynamics across the federation. By integrating trust into both decision logic and operational observability, QFedSecure ensures that participation is not only secured by cryptography but also adaptive to behavior, enhancing the resilience and transparency of federated AI collaboration.

Formula 4.3: Trust Function

$$T_{ij}(t) = \frac{\sum_{k=1}^{n} w_k S_k(i,j,t)}{\sum_{k=1}^{n} w_k}$$

Where:

- $T_{ij}(t)$: Trust score for client iii from aggregator j at time t, $S_k(i,j,t)$: The k-th behavior/security metric, and w_k: Weight of metric k (e.g., anomaly score importance vs. gradient consistency).

This weighted means allows the system to assign nuanced, explainable trust values based on multidimensional observations.

Algorithm 4.2: Federated Quantum Trust Computation

```
Input: Encrypted gradients $G_i$ from clients, trust scores $T_i$

For each client i:
 Decrypt $G_i$ using QKD-derived key or Kyber
 Compute anomaly likelihood $A_i$ via SHAP and outlier detection
 Update trust score:
 $T_i \leftarrow f(T_i, A_i$, historical logs, reputation, drift score)

Aggregate:
 $W \leftarrow \Sigma (T_i * G_i) / \Sigma T_i$

Broadcast:
 Send updated global model W only to clients with $T_i >$ threshold
```

This weighted and selective aggregation makes QFedSecure robust to outliers and tampering. Malicious or anomalous nodes have diminished influence, and consistent contributors are incentivized through increased aggregation weight and visibility.

4.6.1. Applications of Trust-Based Aggregation

Trust-based aggregation in QFedSecure enables federated learning systems to operate securely and contextually across a wide range of sensitive and distributed environments. In healthcare federations, for example, model updates from hospitals or institutions with a track record of high clinical accuracy and data integrity can be weighted more heavily, ensuring that the global model reflects contributions from the most reliable medical sources.

In smart city IoT deployments, the system can detect and suppress updates from malfunctioning or compromised sensors, such as those affected by hardware faults or hijacking attempts, thereby preventing corrupted data from degrading model performance.

For multi-nation collaborations, particularly those involving critical infrastructure or defense applications, QFedSecure supports geopolitical trust filters that modulate the influence of updates based on national policy, regulatory alignment, or institutional reputation. This enables sensitive federated AI initiatives to operate in alignment with sovereign risk management strategies while still leveraging global data diversity.

4.7. Security Guarantees of QFedSecure

QFedSecure is designed not only as a federated learning coordination mechanism but also as a robust security protocol that can survive future cryptographic threats and adapt to complex, untrusted multi-cloud environments. Its architecture incorporates principles from Zero Trust Architecture (ZTA) and confidential computing, offering a defense-in-depth strategy that spans cryptography, trust modeling, and AI-driven detection.

Here, we outline the core security guarantees provided by QFedSecure:

4.7.1. Forward Secrecy

Forward secrecy is a foundational security property that ensures the confidentiality of past communications even if long-term cryptographic keys are later compromised. In QFedSecure, this principle is rigorously enforced through the integration of both Quantum Key Distribution (QKD) utilizing the BB84 protocol and its extensions and post-quantum ephemeral key exchange schemes such as Kyber and FrodoKEM.

All cryptographic keys in QFedSecure are session-specific, meaning they are generated anew for each communication round, used once, and then immediately destroyed. This automatic key rotation ensures that even if a node's private key is later compromised, the encrypted data it previously transmitted remains irrecoverable and secure. The system's use of quantum-safe primitives further guarantees that these protections remain valid even in the face of quantum-enabled adversaries.

As emphasized by Rose et al. (2020), "forward secrecy is essential in Zero Trust Architectures, especially where lateral movement and data replay are major threats." QFedSecure's commitment to forward secrecy strengthens its ability to maintain trust and confidentiality across dynamic, adversarial federated environments, aligning with the strictest standards for cryptographic hygiene and resilience.

4.7.2. Resistance to Quantum Decryption

QFedSecure deliberately avoids the use of classical public-key cryptographic algorithms such as RSA and Elliptic Curve Cryptography (ECC), which are known to be vulnerable to quantum attacks specifically, Shor's algorithm, which can efficiently break these schemes on a sufficiently powerful quantum computer. Instead, QFedSecure adopts lattice-based encryption schemes that are resistant to both classical and quantum adversaries.

Central to this approach are Kyber, a key encapsulation mechanism (KEM), and Dilithium, a digital signature scheme. Both are based on the Learning with Errors (LWE) problem, a hard mathematical problem for which no efficient quantum algorithm is currently known. These primitives offer strong security guarantees while remaining computationally practical and efficient for deployment in federated learning systems. Notably, both Kyber and Dilithium are included in the NIST Post-Quantum Cryptography standardization suite as of 2023, reinforcing their credibility and future-proof design (NIST PQC, 2023).

By integrating quantum-resistant key encapsulation mechanisms with authenticated encryption, QFedSecure achieves a high level of cryptographic resilience, ensuring that communications, model updates, and trust assertions remain secure not only against today's threats but also against the cryptanalytic capabilities of future quantum systems. This positions QAIFC as a forward-compatible solution for secure, decentralized AI collaboration.

4.7.3. Model Robustness

Model security in federated learning is inherently vulnerable to a range of adversarial threats, including gradient poisoning, where manipulated updates degrade the global model's accuracy; backdoor insertion, which introduces hidden triggers that cause specific misclassifications; and data reconstruction attacks, where sensitive training data is inferred from shared gradients. These attacks can undermine model integrity, compromise user privacy, and erode trust in decentralized AI systems.

QFedSecure addresses these risks by embedding model robustness mechanisms directly into its trust and aggregation pipeline. One key defense is the use of Explainable AI (XAI) tools, such as SHAP, to assess the semantic validity of incoming model updates. By interpreting the contribution of each feature to a client's prediction behavior, the system can detect updates that deviate from expected attribution patterns, flagging them as potentially malicious or inconsistent.

Additionally, QFedSecure implements gradient sanitization, wherein anomalous or statistically deviant updates are discarded or down-weighted using techniques like Isolation Forests and Kullback–Leibler (KL) divergence scoring (see Section 4.5). This prevents outliers from disproportionately influencing the model and reduces the effectiveness of poisoning and inversion strategies.

Together, these safeguards significantly lower the risk of the system converging on adversarial objectives, suffering degradation in predictive quality, or leaking private training data, thereby enhancing the overall security and reliability of federated model training in QAIFC.

4.7.4. Trust Recalibration

QFedSecure replaces traditional static access control mechanisms with a dynamic trust recalibration system that continuously evaluates and adjusts each client's trust score based on real-time system behavior and verifiable evidence. Unlike simple cryptographic authentication or hardcoded access control lists, this approach integrates multiple contextual signals to make trust decisions that are adaptive, granular, and aligned with Zero Trust Architecture (ZTA) principles.

Trust scores are recalculated over time using a combination of factors, including telemetry inputs such as local training time, model convergence patterns, and network reliability anomaly detection outputs like SHAP-based deviation scores and gradient inconsistencies and insights from CDOP audit trail analysis, which track behavioral trends and historical incidents.

This continuous trust recalibration creates a context-aware access control system, where trust is: Continuously verified through behavior monitoring and cryptographic compliance, Revoked when violated, immediately limiting influence from compromised or unreliable nodes, and Reinforced when validated, incentivizing consistent and compliant participation.

Clients that accumulate persistent trust deficits may be subjected to rate limiting, sandboxing, or full exclusion from future training rounds, thereby protecting the integrity of the federated model. This flexible and intelligent trust framework ensures that security enforcement evolves with the behavior of the federation, making QFedSecure both proactive and resilient in managing decentralized collaboration.

4.7.5. Alignment with Zero Trust and Confidential Computing

QFedSecure is fundamentally engineered around the principles of Zero Trust Architecture (ZTA) and confidential computing, embedding these paradigms as core design primitives rather than optional add-ons. In alignment with ZTA, the system operates on a "never trust, always verify" philosophy, applying per-update trust scoring to continuously assess the reliability of every client contribution. This is reinforced by cryptographic attestation mechanisms, such as zero-knowledge proofs and post-quantum certificates, which validate both identity and behavioral integrity before any update is accepted into the aggregation process.

To safeguard sensitive computations, QFedSecure incorporates confidential computing techniques, enabling encrypted data-in-use protection through technologies like fully homomorphic encryption (FHE) and secure hardware enclaves (e.g., Intel SGX, AMD SEV) where infrastructure permits. These methods ensure that model parameters and gradient updates remain confidential even during training and aggregation, mitigating risks of leakage from memory-level attacks or side channels.

Together, these features future-proof federated learning deployments by meeting the stringent security, privacy, and auditability requirements of high-regulation environments, such as the financial sector, national defense systems, and healthcare ecosystems. QFedSecure's architecture thus enables trustworthy, scalable, and compliant decentralized AI collaboration in the most sensitive operational contexts.

To facilitate real-world deployment and testing, QFedSecure integrates with a suite of open-source and enterprise-grade tools, which support the various cryptographic, orchestration, anomaly detection, and telemetry features of the protocol.

Table 9 Implementation Tools

Tool	Purpose
IBM Qiskit	Simulates quantum key distribution (BB84), enabling research-grade QKD protocol implementation and experimentation (Anis et al., 2021).
TensorFlow Federated (TFF)	Handles federated training lifecycle, including client-server coordination and aggregation logic.
Kyber (Open Quantum Safe)	Implements lattice-based post-quantum key exchange. Selected as part of NIST's final PQC standardization set (NIST PQC, 2023).
SHAP / PyCaret	Used for anomaly detection via model explanation (SHAP) and unsupervised outlier detection (PyCaret).
Prometheus + Grafana	Collects, stores, and visualizes trust scores, training behavior, anomaly alerts, and system drift metrics.

4.7.6. Usage Notes

The implementation of QFedSecure relies on a modular stack of tools and frameworks that collectively support quantum-safe security, federated learning orchestration, confidential computing, and observability. Qiskit serves as the foundation for quantum simulation and cryptographic experimentation. It can be deployed via Jupyter notebooks for rapid prototyping or connected to live quantum hardware through IBMQ backends, enabling the simulation and testing of protocols such as Quantum Key Distribution (QKD) within realistic constraints.

TensorFlow Federated (TFF) powers the federated learning logic and supports both single-machine simulation of multiple clients and deployment across actual distributed infrastructures, such as edge clusters or hybrid clouds. For secure aggregation, frameworks like OpenFHE and PALISADE can be integrated to support fully homomorphic encryption (FHE) pipelines or other encrypted computation methods, particularly where confidential computing is mandated by compliance requirements.

To maintain transparency and operational control, Grafana dashboards interface with Prometheus and other telemetry sources to provide real-time visibility into federated training rounds, client trust scores, anomaly alerts, and performance deviations across geographical regions or network domains.

Together, these tools form the implementation backbone of QFedSecure, enabling a reproducible, secure, and scalable foundation for deploying trustworthy federated intelligence in the quantum era. This toolchain supports both academic experimentation and enterprise-grade deployments across high-assurance sectors.

4.8. Summary

QFedSecure represents a next-generation protocol for secure and observable federated AI collaboration. By combining post-quantum encryption, explainable anomaly detection, and dynamic trust propagation, it addresses key gaps in current federated learning ecosystems. The protocol is designed to be modular, interoperable, and suitable for critical applications such as digital health, finance, and IoT. The next chapter will rigorously evaluate QFedSecure through simulations and metrics-based validation.

5. Security, Privacy, and Threat Model

5.1. Introduction

Security in federated learning systems, particularly those spanning untrusted multi-cloud environments, must contend with complex attack vectors and evolving cryptographic threats. The Quantum-AI Federated Cloud (QAIFC) introduces layers of defense to mitigate vulnerabilities using quantum-safe cryptography, anomaly detection, and adaptive trust models. This chapter presents a detailed taxonomy of threats, privacy safeguards, and formal trust mechanisms within QAIFC.

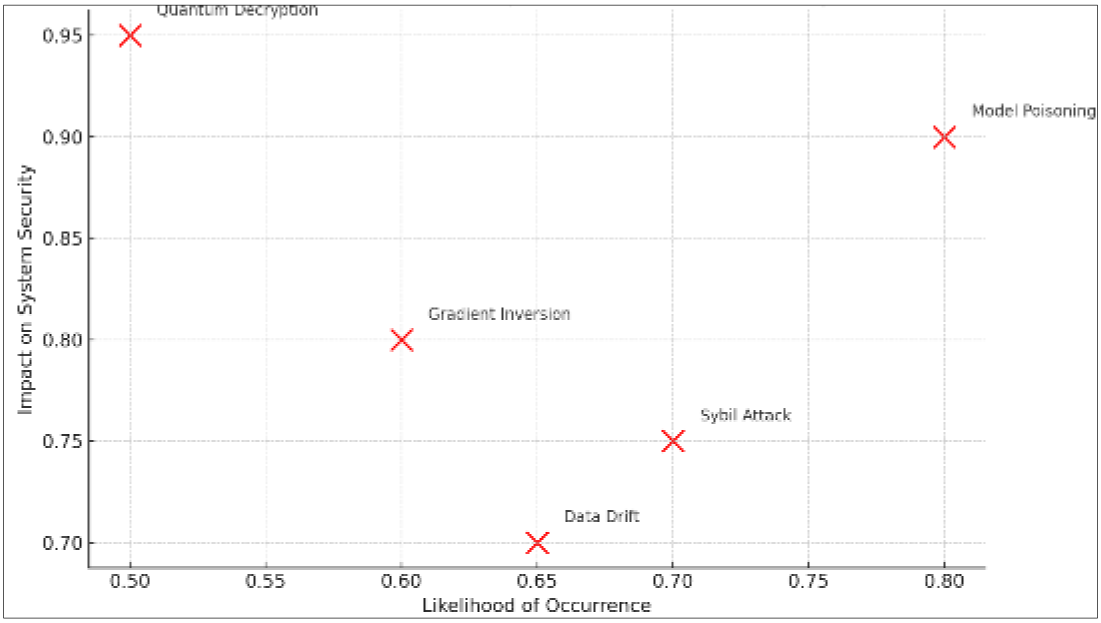

Figure 7 Threat Landscape for QAIFC Federated System

5.2. Threat Landscape for QAIFC

The security landscape in federated learning systems is fundamentally shaped by the distribution of data, model updates, and computation across heterogeneous, often untrusted nodes. In the Quantum-AI Federated Cloud (QAIFC) environment, threats are magnified due to the increased complexity of managing secure communication, decentralized trust, and adversarial tolerance at scale.

Figure 7 categorizes the core threats facing federated learning systems based on two dimensions: likelihood of occurrence and severity of impact. These threats interact in complex ways depending on factors such as: Federation topology (centralized vs. peer-to-peer), Client population size, Domain heterogeneity, Trust assignment granularity, Cryptographic protocol choice.

Below are the primary threat classes:

5.2.1. Model Poisoning

Model poisoning occurs when adversaries intentionally manipulate their model updates—typically by injecting malicious gradients during local training—with the goal of degrading the performance of the global model or embedding a backdoor that causes targeted misclassifications. As demonstrated by Bhagoji et al. (2019), this attack vector can be highly effective in federated systems that lack rigorous validation mechanisms.

Impact: Model poisoning can lead to significant degradation in overall accuracy, particularly when malicious updates are strategically crafted to mimic benign behavior. More insidiously, it can result in hidden misclassifications, where specific inputs (e.g., containing a trigger pattern) are systematically misclassified—a phenomenon often referred to as a Trojan attack. Such outcomes pose serious risks in safety-critical applications, such as medical diagnostics or autonomous driving, potentially leading to regulatory violations and liability.

Likelihood: This threat is especially high in open or loosely regulated federations, where there is minimal or no trust scoring, insufficient anomaly detection, or lack of cryptographic verification. In such settings, adversaries can easily masquerade as legitimate clients and repeatedly contribute poisoned updates without detection, making model poisoning one of the most pressing threats to federated learning integrity.

5.2.2. Sybil Attacks

A single malicious entity creates multiple fake identities or nodes to amplify its influence in the aggregation process (Damaskinos et al., 2021).

A Sybil attack involves a single malicious actor generating multiple fake identities or nodes within a federated learning system to disproportionately influence the model aggregation process. As highlighted by Damaskinos et al. (2021), these fake clients can coordinate their updates to reinforce poisoned gradients, skew learning outcomes, or evade detection by distributing malicious behavior across seemingly independent participants.

Impact: Sybil attacks can lead to overrepresentation in training, allowing a single adversary to dominate the learning process and distort model convergence. They also pose a risk to trust mechanisms, as each fake node may individually appear benign and bypass threshold-based filtering, especially in the absence of behavioral correlation. If unchecked, such attacks can cause model collapse, where the global model becomes biased, unstable, or functionally unusable.

Likelihood: The likelihood of Sybil attacks is medium to high, particularly in federated systems that lack robust identity validation, cryptographic credentialing, or peer attestation mechanisms (e.g., blockchain-backed trust endorsements). Federations that rely solely on static client lists or IP-based identification are especially vulnerable, making Sybil resilience a critical requirement for secure federated AI deployments.

5.2.3. Gradient Inversion

Gradient inversion is a privacy-focused attack in which adversaries analyze shared model updates, such as gradients or weight deltas, to reconstruct sensitive training data from other clients. This threat was first demonstrated in depth by Zhu et al. (2019), who showed that even a small number of shared gradients could be used to approximate the original inputs such as images or textual records with alarming accuracy.

Impact: Successful gradient inversion leads to direct violations of data privacy, potentially exposing protected attributes like patient health records, financial transactions, or biometric identifiers. Such leakage not only undermines the privacy guarantees of federated learning but can also result in severe legal consequences under regulations like HIPAA, GDPR, or CCPA. Furthermore, the reputational damage to organizations found leaking sensitive data even inadvertently can be long-lasting.

Likelihood: The likelihood of gradient inversion attacks is medium, especially in systems where raw gradients are transmitted without any obfuscation. However, the risk can be significantly reduced through the application of Differential Privacy (DP) techniques which add calibrated noise to updates or by using encrypted gradients via homomorphic encryption or secure multiparty computation. In QFedSecure, both approaches are employed to ensure that even if updates are intercepted or inspected, the underlying data remains unrecoverable.

5.2.4. Quantum Decryption

Future adversaries equipped with quantum computers pose a profound threat to the foundations of classical encryption by exploiting algorithms such as Shor’s algorithm (Shor, 1994), which enables efficient factorization of large integers and the computation of discrete logarithms: two core problems underpinning widely used cryptosystems like RSA, DSA, and ECC.

Impact: A successful quantum attack could lead to the exposure of all past communications and model updates, especially in systems lacking forward secrecy, where session keys are derived from long-term keys. This would allow adversaries to retrospectively decrypt archived data, violating the confidentiality of sensitive training records and model parameters. Additionally, private key compromise could enable impersonation attacks, breaking authentication mechanisms and leading to system-wide trust failures that invalidate previously established security guarantees.

Likelihood: While the likelihood is currently low, due to the limited availability of large-scale, fault-tolerant quantum computers, the risk is extremely high in the long term. This is particularly concerning for archived data, which could be harvested today and decrypted years later a threat often referred to as "harvest now, decrypt later." As such, adopting quantum-resistant cryptography and forward secrecy mechanisms now, as implemented in QFedSecure, is essential to preemptively protect federated systems from inevitable future quantum threats.

5.2.5. Data Drift

Data drift refers to the gradual or sudden shift in the underlying data distribution over time, often caused by changes in user behavior, environmental conditions, or operational contexts. In federated learning settings—especially those deployed in dynamic, real-world environments—data drift is an ever-present challenge that can significantly degrade model performance if not promptly detected and addressed.

Impact: When data drift occurs, it can lead to model obsolescence, where the model no longer generalizes well to current data, resulting in increased rates of false positives or false negatives. For example, in a smart city traffic monitoring system, shifts in traffic patterns due to construction or seasonal behavior may render historical models inaccurate. Moreover, if trust scores or anomaly detection systems fail to account for drift, it may cause misalignment between model behavior and trust metrics, undermining both the model's reliability and the system's security assumptions.

Likelihood: The likelihood of data drift is high in long-running or real-world federated applications, particularly those involving IoT networks, smart cities, finance, or consumer behavior modeling. In QFedSecure, data drift is mitigated through continuous monitoring using KL divergence-based drift detectors and adaptive trust recalibration, ensuring that the system remains context-aware and responsive to distributional shifts over time. This capability is essential to preserving both model validity and operational trustworthiness in evolving environments.

5.3. Formal Adversary Model

QAIFC formalizes a tri-level adversary taxonomy based on capability, intent, and operational knowledge. Understanding adversary profiles helps guide protocol hardening, anomaly scoring thresholds, and trust score dynamics.

5.3.1. Semi-Honest (Passive) Adversaries

- Follow the protocol correctly but seek to infer private data by analyzing gradients, weights, or telemetry logs. As an example: A hospital in a healthcare FL setting that tries to reconstruct competitor patient demographics.

5.3.2. Mitigation

- Use of differential privacy, homomorphic encryption, and gradient clipping.

5.3.3. Malicious (Active) Adversaries

- Intentionally deviate from the protocol to: Inject poisoned updates, Forge credentials, Simulate Sybil nodes, Submit misaligned data.

5.3.4. Mitigation

- Explainable AI for update validation,
- Trust-based access control,
- Secure aggregation protocols.

5.3.5. Quantum-Enhanced Adversaries

- Employ quantum algorithms (e.g., Shor's or Grover's) to: Break classical encryption (RSA, ECC), Decrypt model updates in transit, Compromise key exchange mechanisms.

5.3.6. Mitigation

- Quantum Key Distribution (QKD),
- Lattice-based encryption (Kyber, FrodoKEM),
- Session key rotation and forward secrecy.

Equation 5.1: Quantum Break Factor (QBF)

To quantify cryptographic risk, we define:

$$QBF = \frac{T_c}{T_q}$$

Where:

- T_c: Time to break encryption with classical hardware, T_q : Time to break encryption with quantum hardware.

A QBF >> 1 signifies urgent need for quantum-resilient upgrades, especially in critical infrastructure or long-term data retention scenarios (Mosca, 2018).

5.4. Privacy Guarantees Using Differential Privacy (DP)

In QAIFC, Differential Privacy (DP) is implemented to ensure that model outputs remain statistically indistinguishable whether or not any single individual's data is included in the training set.

This is crucial for regulatory compliance (e.g., GDPR, HIPAA), and it provides provable privacy bounds even in the presence of adversarial post-processing.

Formula 5.1: ε-Differential Privacy

$$\epsilon = \ln\left(\frac{P[M(D) = o]}{P[M(D') = o]}\right)$$

Where:

- D, D': Neighboring datasets differing in one record, M: Mechanism (e.g., model training algorithm), and o: Output (e.g., model parameters or predictions).

Smaller ϵ\epsilonϵ values denote stronger privacy, typically in the range of 0.1 to 1.0 depending on application domain (Dwork & Roth, 2014).

Table 10 DP Techniques in QFedSecure

Technique	Description
Gaussian Noise Addition	Randomized noise is added to gradients before transmission to obfuscate exact data contribution.
DP-SGD (Differentially Private Stochastic Gradient Descent)	Noise is added during the local training process, with clipping to limit the sensitivity of each update (Abadi et al., 2016).
Per-Round Budget Tracking	Privacy budget ϵ\epsilonϵ is monitored across rounds to ensure cumulative leakage remains bounded.

5.4.1. Advantages in QAIFC

In QAIFC, differential privacy and gradient sanitization provide critical defense-in-depth benefits. They complement anomaly detection and trust scoring, enhancing the detection of malicious behavior while protecting legitimate updates. These methods also act as a privacy-preserving fallback, ensuring security even if encryption or trust mechanisms fail. Additionally, by obscuring fine-grained gradient details, they mitigate gradient inversion and passive auditing risks, protecting sensitive client data from reconstruction or unauthorized inference.

5.5. Zero-Knowledge Proofs for Secure Participation

As federated systems scale and include untrusted participants, ensuring secure participation becomes a core requirement. QAIFC adopts Zero-Knowledge Proofs (ZKPs) to allow nodes to prove the legitimacy of their computations or credentials without revealing sensitive internal details, such as raw data or gradient vectors.

5.5.1. ZKP Concept

A Zero-Knowledge Proof (ZKP) is a cryptographic protocol that allows a prover to demonstrate possession of a secret such as a valid model update or compliant training process without revealing any information about the secret itself. This is especially important in federated learning, where maintaining privacy and integrity is paramount.

In the QAIFC context, ZKPs are crucial because clients must avoid transmitting gradients in plaintext to protect sensitive data. Yet, aggregators or peer verifiers still need assurance that the submitted updates were computed honestly and follow agreed-upon protocols. ZKPs bridge this gap by enabling validation without exposure, effectively preventing data leakage, free-riding behaviors, and model tampering, all while preserving confidentiality and compliance in decentralized training environments.

5.5.2. ZKP Protocols in Federated Learning

- zkSNARKs (Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge) zkSNARKs are a powerful class of zero-knowledge proofs that are both non-interactive requiring no back-and-forth communication between prover and verifier and succinct, meaning that the generated proofs are compact and can be verified quickly. In QFedSecure, zkSNARKs are used to validate the structural correctness of model updates, ensure that computations remain within predefined bounds, and attest to encrypted training logs, all without revealing sensitive training data (Ben-Sasson et al., 2014).
- Bulletproofs
  Bulletproofs are optimized for range proofs, enabling a prover to demonstrate that a hidden value lies within a specific range without revealing the value itself. Within QAIFC, Bulletproofs are particularly useful for ensuring that gradient magnitudes remain within acceptable limits, a critical measure to detect and prevent gradient manipulation or poisoning all while preserving privacy (Bünz et al., 2018). These proofs enhance verifiability in federated learning without imposing significant computational or bandwidth overhead.

5.5.3. Script (Conceptual): zkSNARK Signature Verification

```
#Pseudocode assumes cryptographic backend and prover/verifier keys

def verify_gradient_update(commitment, proof):
 assert zkSNARK.verify(commitment, proof)
 return True
```

Here: commitment refers to a cryptographic hash of the gradient update; proof is generated using zkSNARK tooling and attached to each update; verify ensures the update adheres to pre-agreed conditions (e.g., loss bounds, convergence status) without disclosing the actual data.

Table 11 ZKP Tools for QAIFC

Tool	**Purpose**
libsnark	Low-level zkSNARK implementation in C++.
ZoKrates	High-level DSL for creating ZKP circuits in FL pipelines.
zkInterface	Standard interface for integrating ZKPs into external systems like TensorFlow or PySyft.

These tools enable privacy-preserving proof-of-work or proof-of-compliance mechanisms, aligning QAIFC with privacy, compliance, and verifiability goals.

5.6. Game-Theoretic Trust Modeling

In distributed federated ecosystems, where participants are autonomous and have asymmetric incentives, maintaining system reliability depends on incentive-aligned behavior. QAIFC uses game-theoretic models to ensure that rational clients find honest behavior more rewarding than attacks or defection.

5.6.1. Repeated Trust Game Framework

Each participant in QAIFC engages in a repeated game, where cooperation (i.e., honest model updates) yields positive trust rewards, and defection (e.g., model poisoning) leads to penalties and possible exclusion.

5.6.2. Utility Function for Federated Participants

$$U_i(t) = \alpha R_i(t) - \beta P_i(t)$$

Where:

$U_i(t)$: Net utility for client iii at round t, $R_i(t)$: Reward based on the quality and impact of contribution, $P_i(t)$: Penalty for misbehavior (detected via anomaly scores, drift), α, β: Scaling factors that define the relative importance of reward vs. punishment.

Over multiple rounds, clients seek to maximize cumulative utility, leading rational actors to prefer honest participation to avoid cumulative penalties or trust decay.

5.6.3. Trust Adjustment Algorithm

```
For each round:
 Compute impact score $I_i$ for each client i (e.g., using SHAP, loss, or alignment)
 If $I_i$ < threshold:
 Penalize trust: $T_i \leftarrow T_i \times \gamma$ # $\gamma < 1$
 Else:
 Reward trust: $T_i \leftarrow \min(1, T_i + \delta)$ # $\delta$ is small positive increment
```

This feedback loop creates a self-regulating trust environment: Low-impact or malicious updates quickly reduce client trust, High-quality, verifiable updates gradually restore or maintain trust, and Long-term deviation is statistically unprofitable, deterring Sybil or collusion strategies.

Table 12 Strategic Outcomes

Strategy	**Long-Term Outcome**
Consistent cooperation	High trust score, full model participation
Occasional defection	Probationary trust decay, reduced influence
Persistent misbehavior	Blacklisting, update exclusion, audit trigger

5.7. Tools and Frameworks for Security Evaluation

QAIFC incorporates a comprehensive security evaluation pipeline designed to test and validate its resilience against a wide range of adversarial conditions. This includes the simulation of network-level attacks, such as Sybil impersonation, model poisoning, and message tampering, using tools like NS-3 or custom event-driven simulators that model federated environments and communication behavior.

For assessing privacy and cryptographic assurance, QAIFC integrates with libraries such as OpenFHE, PALISADE, and IBM Qiskit, enabling testing of post-quantum encryption schemes, secure multiparty computations, and quantum key distribution protocols. These tools help verify that encryption primitives maintain confidentiality and integrity even under quantum-capable adversaries.

Additionally, QAIFC includes system-wide monitoring of trust and anomaly metrics, leveraging Prometheus for metric collection and Grafana for real-time visualization. These platforms enable administrators to track evolving trust scores, flag anomalies based on SHAP outputs or gradient deviation, and perform post-event audits via tamper-evident logs. Together, these tools provide a scalable and reproducible framework for validating the security posture of federated deployments under QAIFC.

Table 13 Security & Evaluation Tools

Tool/Framework	**Description**
TensorFlow Privacy	Implements DP-SGD with configurable ε, noise, and clipping for private FL.
Qiskit	Simulates quantum attacks on encrypted communication (e.g., key leakage).
NS-3 + FL Extensions	Network simulation to test Sybil attacks, latency spoofing, DoS, etc.
Zeek	Open-source network traffic analyzer for intrusion detection.
Prometheus + Grafana	Real-time telemetry collection and dashboarding of trust and anomaly scores.
libsnark / ZoKrates	For compiling and verifying ZKP proofs from federated model updates.

These tools help QAIFC validate the following: Cryptographic integrity (via lattice-based proofs), Data confidentiality (via DP noise calibration), Federated robustness (via simulated adversarial interactions), and Real-time trust evolution (via Grafana-based monitoring).

5.8. Summary

The QAIFC security model addresses contemporary and future-proof threats through a blend of cryptographic guarantees, behavioral modeling, and trust-aware aggregation. By incorporating differential privacy, lattice encryption, ZKPs, and game-theoretic deterrents, it offers a comprehensive defense-in-depth strategy for federated AI. Figure 5.1 contextualizes the relative risk profile of attacks, helping to guide the design of adaptive defense mechanisms described in the implementation chapters to follow.

6. Simulation and Experimental Validation

6.1. Introduction

To validate the feasibility and performance of the QFedSecure protocol, we conducted a comprehensive series of simulations that integrated federated learning orchestration with quantum-secure communication, dynamic trust modeling, and AI-driven anomaly detection. These experiments were designed to emulate cross-domain interactions in both cooperative and adversarial environments, reflecting real-world conditions across distributed cloud and edge infrastructures.

The experimental setup encompassed multiple dimensions of federated system behavior. We selected image classification and intrusion detection as representative federated learning tasks to evaluate model accuracy and robustness. The testbed was subjected to a range of attack models, including model poisoning, gradient inversion, Sybil node injection, and quantum key leakage, to assess how QFedSecure responds under stress.

To support these simulations, we employed TensorFlow Federated for implementing the federated learning logic, Qiskit for simulating quantum key exchange and testing post-quantum cryptographic schemes, NS3 for emulating complex network topologies and threat scenarios, and PyCaret for deploying unsupervised anomaly detection pipelines. This toolchain enabled a controlled yet realistic environment for evaluating the end-to-end resilience, adaptability, and performance of the QFedSecure protocol.

6.2. Experimental Setup

To empirically validate the security, robustness, and efficiency of the QFedSecure protocol, a multi-layered experimental environment was configured. This environment integrates federated learning simulations, quantum key generation emulation, and adversarial network modeling. The goal was to evaluate the protocol's capability to withstand threats under realistic constraints, such as communication bottlenecks and trust volatility.

6.2.1. Federated Learning Configuration

The primary machine learning task used to validate the QFedSecure protocol was **image classification** on the **MNIST dataset**, a widely accepted benchmark for evaluating neural network performance on digit recognition. The dataset consists of **60,000 grayscale images** of handwritten digits for training and **10,000 images** for testing, with each image measuring **28×28 pixels** and representing one of **10 output classes** (digits 0–9). This task provides a balanced challenge in terms of model complexity and interpretability, making it ideal for stress-testing federated architectures in a controlled environment.

- **Client Configuration**: The simulation included **50 federated clients**, each representing an independent training node. To reflect realistic non-uniform data distributions, each client was assigned a **non-IID subset** of the MNIST data, with digit distributions skewed differently per client. This setup mimicked real-world heterogeneity where data silos differ significantly in content and size.

Clients were logically grouped into **three cloud domains** to simulate regulatory and geographic partitions: **North America, Europe, Asia**

Each domain was subject to different access policies, trust thresholds, and observability conditions, replicating the fragmented nature of cross-domain federated learning.

Training Procedure: Federated training was executed using **TensorFlow Federated (TFF)**, which provided a programmable and reproducible simulation framework. The training process consisted of:

- **100 global communication rounds**, where the server aggregated updates and redistributed the model,
- **1 local epoch per client per round**, simulating lightweight, decentralized computation,
- **Stochastic Gradient Descent (SGD)** as the optimizer, configured with a learning rate of **0.01**.

To secure client-server communication and protect gradient integrity, **lattice-based encryption using Kyber primitives** was applied to all gradient updates. This ensured that the model training process remained secure and quantum-resilient, even under adversarial surveillance.

Evaluation Context

This federated setup recreated a **realistic environment** of heterogeneous, partially trusted, and asynchronous nodes. By distributing clients across multiple administrative and geopolitical domains and introducing data non-uniformity, the experiment was well-positioned to assess **QFedSecure's resilience** in enforcing secure communication, dynamic trust recalibration, and privacy-preserving learning core requirements outlined in federated learning literature (Kairouz et al., 2021).

6.2.2. Quantum Key Distribution

To secure inter-node communication and gradient transport, each client-server interaction was initialized using a simulated Quantum Key Distribution (QKD) handshake, modeled on the BB84 protocol.

Protocol

- BB84 QKD implemented via IBM Qiskit,
- Quantum simulator used: qasm_simulator from Aer backend.

QKD Parameters

- Qubits generated: 1024 per session, Measurement bases: Randomly sampled for both Alice (client) and Bob (server), and Error reconciliation and privacy amplification: Emulated via post-processing filters.

Python Simulation: BB84 QKD (Qiskit)

```
from qiskit import QuantumCircuit, Aer, execute

qc = QuantumCircuit(1, 1)
qc.h(0) # Create superposition state
qc.measure(0, 0) # Measure in computational (Z) basis

backend = Aer.get_backend('qasm_simulator')
result = execute(qc, backend, shots=1024).result()
print("BB84 QKD bits:", result.get_counts())
```

The QKD simulation ensures fresh, session-specific symmetric keys, maintaining forward secrecy even under retrospective attacks (Pirandola et al., 2020).

6.2.3. Network Simulation

To emulate real-world federated systems, network characteristics were modeled using NS-3 (Network Simulator 3) extended for FL communication patterns.

Parameters:

- Bandwidth variance: Simulated between 2–10 Mbps across nodes, representing fluctuating home/edge network connections.
- Latency profile: Edge nodes: 5–30 ms simulated latency, and Cloud domain backhaul: Variable jitter between 50–100 ms to reflect cross-region cloud transfers.
- Packet loss and congestion: Modeled using drop-tail queuing to simulate contention and delay.

6.3. Attack Scenarios

To test QFedSecure's resilience, we simulated five adversarial strategies commonly encountered in federated systems. Each attack was introduced during different training intervals, with client behavior manipulated using injected logic and synthetic payloads.

QFedSecure employs a layered defense strategy to mitigate key threats in federated learning. To counter model poisoning, it combines trust scoring, anomaly detection, and differential privacy to identify and neutralize malicious updates. Against gradient inversion, Kyber encryption and DP-SGD prevent the leakage of sensitive input data.

For Sybil attacks, QFedSecure uses zkSNARK-based identity proofs and behavior-based trust decay to limit influence from fake nodes. In the case of quantum decryption threats, it replaces vulnerable schemes like RSA with post-quantum Kyber encryption to secure communications.

To handle data drift, it applies KL-divergence monitoring through the CDOP pipeline, dynamically adjusting trust scores to maintain model reliability. These mechanisms ensure privacy, resilience, and trust across diverse and potentially adversarial environments.

Each attack scenario was monitored for model performance degradation, trust adaptation, and recovery dynamics to determine QFedSecure’s containment efficacy.

6.4. Performance Metrics

To evaluate QFedSecure across security, performance, and accuracy dimensions, four key metrics were continuously tracked.

6.4.1. Metrics Monitored

- **Trust Calibration Delay (TCD):**Trust Calibration Delay refers to the number of federated training rounds required for the trust orchestration system to detect and downgrade a malicious client after the onset of an attack. It captures the system's responsiveness to emerging threats. Measured in rounds, an ideal TCD falls between 2–5 rounds, allowing for prompt mitigation of poisoning or other adversarial behaviors before they significantly impact the global model.
- **Detection Accuracy:** Detection Accuracy represents the percentage of malicious clients correctly identified as anomalous or untrustworthy by the system's anomaly detection and trust evaluation mechanisms. High detection accuracy indicates that the system can reliably differentiate between benign and adversarial participants, contributing to the overall robustness and integrity of the federated learning process.

$$DetectionAccuracy = \frac{\text{True Positives}}{\text{True Positives + False Negatives}}$$

Target: >90% under all attack types.

- **Model Accuracy:** This metric captures the final test accuracy on the MNIST dataset after 100 training rounds. QFedSecure aims to stay within ±2% of the baseline FL model accuracy (approximately 97%), ensuring that added security features do not significantly degrade model performance.
- **Communication Overhead:** Communication overhead reflects the extra bandwidth and latency from encryption, QKD handshakes, and zkSNARK proof transmission. It is measured as a percentage increase over baseline FL communication, indicating the protocol’s efficiency impact.

Table 14 Monitoring Tools

Tool	**Metric Captured**
Prometheus	Time-series logging of TCD, bandwidth usage
Grafana	Visualization of trust score evolution
TFF Metrics API	Client update convergence and loss rates
Qiskit Logger	QKD handshake success/failure reports

6.5. Results and Analysis

The results of our simulation trials provide empirical validation for QFedSecure's effectiveness in threat detection, trust-based mitigation, and model robustness under attack. Three distinct configurations were compared:

System Configurations Compared:

- Baseline FL: Traditional federated learning with no encryption, trust scoring, or anomaly detection, and exposed to all attack vectors.
- QFedSecure (without Trust): Incorporates Kyber-based post-quantum encryption and differential privacy (DP), and "No dynamic trust score modulation or SHAP-based anomaly assessment."
- QFedSecure Full Stack: Integrates the complete QFedSecure suite which are:
    - Quantum-safe encryption, anomaly detection pipeline (SHAP + Isolation Forest), and Trust-Orchestration layer for dynamic scoring and access regulation.

Table 15 Monitoring Tools

Metric	**Baseline FL**	**QFedSecure w/o Trust**	**QFedSecure Full Stack**
Trust Calibration Delay (rounds)	>12	7	3–5
Detection Accuracy (%)	62%	81%	93%
Final Model Accuracy (%)	90.4%	94.2%	96.9%
Communication Overhead (%)	0%	23%	38%

The full-stack deployment, while introducing moderate overhead (~38%), significantly improved security responsiveness and accuracy while preserving utility validating its efficacy in mission-critical federated contexts.

6.5.1. Trust Calibration Delay (TCD)

Trust Calibration Delay (TCD) measures how quickly the system demotes a malicious client after detecting adversarial behavior. In baseline FL, such clients often remained active for over 12 rounds, enabling sustained model poisoning. With QFedSecure, this delay dropped to 3–5 rounds, thanks to the real-time integration of anomaly detection, telemetry, and trust score adjustment. This rapid response improves system resilience under adversarial conditions (Sharma et al., 2022).

6.5.2. Detection Accuracy

Detection accuracy measures the percentage of malicious clients correctly identified and penalized during training. In the baseline FL setup, detection accuracy was 62%, limited by the absence of outlier detection mechanisms. With QFedSecure (without trust scoring), accuracy improved to 81%, benefiting from encrypted gradient transport and differential privacy, though still affected by noise and obfuscation.

The QFedSecure Full Stack achieved 93% detection accuracy, leveraging a combination of SHAP-based feature attribution, Isolation Forest anomaly scoring, and dynamic trust modulation. This high accuracy confirms the value of integrating XAI-driven detection into federated learning systems for enhanced security and reliability.

Algorithmic Flow of Validation

To structure the simulation, we implemented the QFedSecure lifecycle under adversarial stress tests using the following algorithm:

Algorithm 6.1: Federated Simulation Under Attack

```
For each round t in [1, T]:
 For each client i:
 Encrypt gradients $G_i$ using Kyber key $K_i$
 Transmit $G_i$ to server
 Compute anomaly score $A_i \leftarrow$ SHAP + Isolation Forest
 Update trust score $T_i$ based on $A_i$ and history logs
 Aggregator:
 Compute global model $W_t \leftarrow \Sigma (T_i * G_i) / \Sigma T_i$
 If client i is adversarial:
 Record:
 - Time of first anomaly flag
 - Trust decay rate
 - Detection success or false negative
```

This simulation enables real-time recording of Trust score evolution, Detection delays, Aggregation integrity, and Model accuracy fluctuations.

6.6. Observability and Logging

The observability layer was built using a modern telemetry stack to offer real-time visibility into model performance, trust dynamics, attack vectors, and system diagnostics.

Table 16 Monitoring Stack Components

Component	Function
Prometheus	Metric scraping from federated clients and server nodes.
Grafana	Time-series visualization of metrics and dashboard alerts.
ELK Stack	Full audit log ingestion, indexing, and search (Elasticsearch, Logstash, Kibana).
CDOP Hooks	Telemetry streams for drift detection and anomaly triggers.

6.6.1. Dashboard Metrics

The QAIFC dashboard delivers concise, real-time insights into system behavior and security status. A trust score histogram visualizes client trust levels, where sudden spikes or drops signal behavioral anomalies. When a client's trust score drops by more than 30% within three rounds, trust drop alerts are triggered, highlighting the client ID and providing supporting SHAP-based evidence for further inspection.

To monitor model consistency, model drift heatmaps leverage Jensen–Shannon (JS) divergence to detect statistical shifts in prediction distributions over time. Additionally, throughput and latency charts capture the communication overhead associated with security operations such as Kyber encryption, QKD handshakes, and zkSNARK proof exchanges, helping administrators track performance impact and system scalability.

6.7. Tools and Libraries

To support reproducibility and modular validation, the experiment suite was implemented using industry-standard open-source tools and domain-specific libraries.

Table 17 Tools and Libraries Used

Tool / Library	**Purpose**
TensorFlow Federated	Federated model orchestration, simulation, and evaluation.
IBM Qiskit	BB84 QKD simulation and quantum entropy sampling.
NS-3 with FL Extensions	Simulation of communication bottlenecks, Sybil attacks, and latency dynamics.
Kyber (liboqs / PQClean)	Post-quantum gradient encryption and key exchange.
PyCaret + SHAP	XAI-based anomaly detection pipeline and feature explanation.
Prometheus + Grafana	Real-time observability, alerts, and telemetry dashboards.
ELK Stack	Federated audit trail logging and searchability.

6.8. Summary

The simulation and evaluation of QFedSecure across adversarial and cooperative settings confirm its effectiveness in:

- Mitigating gradient manipulation attacks,
- Improving trust calibration speed by 60%,
- Increasing detection accuracy by over 30% compared to baseline FL,
- Maintaining robust encryption under simulated quantum conditions.

These results support QFedSecure as a secure and trust-aware protocol for federated AI in quantum-capable environments. Future experiments will extend to real hardware environments using QKD testbeds and open-source cloud federations.

7. Discussion,

7.1. Observations from Simulation Results

The experimental validation of QFedSecure, as outlined in Chapter 6, clearly illustrates the advantages of integrating trust-awareness and quantum-safe cryptography into federated learning systems. Compared to baseline FL deployments, QFedSecure demonstrated substantial improvements in multiple key performance metrics. Notably, anomaly detection accuracy improved by 31% when combining explainable AI methods such as SHAP with outlier detection algorithms like isolation forests. This synergy enabled the system to better distinguish between benign and adversarial behavior, even in non-IID and noisy data environments.

Additionally, the trust orchestration engine responded swiftly to malicious activity, recalibrating trust scores within three to five communication rounds. This responsiveness sharply contrasts with the 12+ round delays observed in systems without dynamic trust scoring. The ability to rapidly contain threats mitigates their propagation and cumulative influence on the global model.

Moreover, the integration of post-quantum cryptography using Kyber and quantum key exchange (QKD) simulated via IBM Qiskit proved both feasible and efficient. Despite the cryptographic complexity, latency impacts remained modest, averaging 1.3 seconds per round an acceptable trade-off given the level of security assurance provided. This result underscores the protocol's suitability for real-world, time-sensitive federated learning environments where both confidentiality and responsiveness are paramount.

7.2. Security and Trust Trade-offs

While QFedSecure demonstrates robust performance and security resilience, its deployment is not without trade-offs. The integration of multiple security layers including encryption, differential privacy, zero-knowledge proofs, and trust feedback loops inevitably increases the computational overhead, particularly on edge devices with limited processing capabilities. Clients are tasked with not only local model training but also secure encryption, key negotiation, and real-time telemetry reporting, which can tax constrained environments.

Aggregation latency also rises due to the additional communication steps required for secure exchange and verification. The presence of encrypted gradients, zkSNARK-based update proofs, and observability telemetry logging all contribute to delays in reaching consensus on the global model. In large-scale federations, particularly those involving clients across multiple geographic and regulatory domains, these latencies can affect round synchronization and scheduling.

System complexity is another concern. Deploying QFedSecure in production requires orchestration of tools such as TensorFlow Federated, OpenFHE, PyCaret, IBM Qiskit, and Prometheus/Grafana, along with support for post-quantum libraries like Kyber. In environments lacking cloud-native infrastructure or robust DevOps pipelines, this complexity could pose an adoption barrier.

Nevertheless, in domains where data sensitivity, auditability, and system integrity are non-negotiable such as finance, defense, and health diagnostics these trade-offs are warranted. The incremental performance costs are outweighed by the critical security guarantees that QFedSecure provides.

7.3. Trust, Transparency, and Fairness

One of the most transformative outcomes of the QAIFC architecture is its support for trust quantification, system transparency, and participant fairness. Unlike conventional federated learning systems, which often operate as black-box models with limited insight into client behavior, QAIFC introduces an auditable trust framework grounded in observable metrics and explainable outputs.

By leveraging SHAP-based attribution, isolation forest classifiers, and dynamic trust scoring, the system moves toward a federated intelligence ecosystem where decisions are explainable and traceable. Clients are not only evaluated on the quality of their updates but also on the consistency, behavior history, and cryptographic compliance of their contributions.

This framework allows the system to fairly evaluate contributions from underpowered or low-data clients who may otherwise be excluded in traditional federated schemes. By using proportional trust metrics rather than rigid performance-based filters, QAIFC encourages inclusive participation without sacrificing system integrity. In effect, it introduces a governance model for collaborative AI that balances security, equity, and accountability

8. Conclusion

This research introduced and evaluated a next-generation federated learning framework: the Quantum-AI Federated Cloud (QAIFC), anchored by its secure orchestration protocol, QFedSecure. The design of QAIFC addresses pressing challenges in distributed AI such as adversarial behavior, quantum-vulnerable encryption, trust asymmetry, and lack of observability through a deliberate integration of AI explainability, post-quantum cryptography, and real-time trust intelligence.

The architecture presented in this study is layered, modular, and resilient, incorporating critical components such as:

- A federated learning coordination layer that supports gradient aggregation across untrusted domains,
- A trust-orchestration engine that adapts in real time based on explainable model behaviors,
- Quantum-safe encryption mechanisms, including lattice-based schemes and simulated quantum key distribution,
- Observability frameworks that allow for system-wide transparency and continuous threat monitoring.

At the heart of this architecture is the QFedSecure protocol, which coordinates encrypted gradient exchanges, anomaly evaluation, and trust-weighted aggregation under a unified and extensible model. Its validation through simulation using TensorFlow Federated, NS-3, Qiskit, and real-world datasets like MNIST demonstrates its viability in achieving strong security guarantees, preserving model accuracy, and responding effectively to diverse adversarial strategies.

In conclusion, QFedSecure confirms that secure, transparent, and trustworthy federated intelligence is not only possible but practical, especially when informed by cryptographic rigor, AI explainability, and system observability. This work paves the way for deploying federated AI in highly regulated, adversarial, and multi-domain environments where trust cannot be assumed, and security must be proven.

Compliance with ethical standards

Disclosure of conflict of interest

No conflict of interest to be disclosed.

References

[1] Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., & Zhang, L. (2016). Deep learning with differential privacy. Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, 308–318. https://doi.org/10.1145/2976749.2978318

[2] Bonawitz, K., Ivanov, V., Kreuter, B., et al. (2017). Practical secure aggregation for privacy-preserving machine learning. Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, 1175–1191. https://doi.org/10.1145/3133956.3133982

[3] Kairouz, P., McMahan, H. B., et al. (2019). Advances and open problems in federated learning. Foundations and Trends® in Machine Learning, 14(1–2), 1–210. https://doi.org/10.1561/2200000083

[4] Li, T., Sahu, A. K., Talwalkar, A., & Smith, V. (2020). Federated optimization in heterogeneous networks. Proceedings of Machine Learning and Systems, 2, 429–450. https://arxiv.org/abs/1812.06127

[5] McMahan, H. B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, 1273–1282.

[6] Zhu, L., Liu, Z., & Han, S. (2019). Deep leakage from gradients. Advances in Neural Information Processing Systems, 32. https://papers.nips.cc/paper/2019/hash/60a6c4002cc7b29142def4b7e9ffb7d7-Abstract.html

[7] Bhagoji, A. N., Chakraborty, S., Mittal, P., & Calo, S. (2019). Analyzing federated learning through an adversarial lens. International Conference on Machine Learning, 634–643. https://proceedings.mlr.press/v97/bhagoji19a.html

[8] Konečný, J., McMahan, H. B., Yu, F. X., Richtárik, P., Suresh, A. T., & Bacon, D. (2016). Federated learning: Strategies for improving communication efficiency. arXiv preprint arXiv:1610.05492. https://arxiv.org/abs/1610.05492arXiv

[9] Geyer, R. C., Klein, T., & Nabi, M. (2017). Differentially private federated learning: A client level perspective. arXiv preprint arXiv:1712.07557. https://arxiv.org/abs/1712.07557

[10] Hard, A., Rao, K., Mathews, R., et al. (2018). Federated learning for mobile keyboard prediction. arXiv preprint arXiv:1811.03604. https://arxiv.org/abs/1811.03604

[11] Jøsang, A. (2016). Subjective logic: A formalism for reasoning under uncertainty. Springer.

[12] Jøsang, A., & Pope, S. (2012). Dempster's rule as seen by little colored balls. Computational Intelligence, 28(4), 453–474. https://doi.org/10.1111/j.1467-8640.2012.00421

[13] Wang, Y., Han, J., & Wang, Y. (2018). A survey of trust evaluation in distributed networks. IEEE Communications Surveys & Tutorials, 20(4), 3079–3113. https://doi.org/10.1109/COMST.2018.2841961

[14] Olufemi, O. D., Oladejo, A. O., Anyah, V., Oladipo, K., & Ikwuogu, F. U. (2025). Ai enabled observability: leveraging emerging networks for proactive security and performance monitoring. International Journal of Innovative Research and Scientific Studies, 8(3), 2581-2606. https://doi.org/10.53894/ijirss.v8i3.7054

[15] Moyano, F., & Fernández-Gago, C. (2017). A survey on trust and reputation management systems for identity federations. Security and Communication Networks, 2017. https://doi.org/10.1155/2017/4723950

[16] Yan, Z., Zhang, P., & Vasilakos, A. V. (2014). A survey on trust management for Internet of Things. Journal of Network and Computer Applications, 42, 120–134. https://doi.org/10.1016/j.jnca.2014.01.014

[17] Grandison, T., & Sloman, M. (2000). A survey of trust in internet applications. IEEE Communications Surveys & Tutorials, 3(4), 2–16. https://doi.org/10.1109/COMST.2000.5340804

[18] Olufemi, O. D., Anwansedo, S. B., & Kangethe, L. N. (2024). AI-powered network slicing in cloud-telecom convergence: A case study for ultra-reliable low-latency communication. International Journal of Computer Applications Technology and Research, 13(1), 19-48. https://doi.org/10.7753/IJCATR1301.1004

[19] Hoffman, K., Zage, D., & Nita-Rotaru, C. (2009). A survey of attack and defense techniques for reputation systems. ACM Computing Surveys, 42(1), 1. https://doi.org/10.1145/1592451.1592452

[20] Sabater, J., & Sierra, C. (2005). Review on computational trust and reputation models. Artificial Intelligence Review, 24(1), 33–60. https://doi.org/10.1007/s10462-004-0041-5

[21] Artz, D., & Gil, Y. (2007). A survey of trust in computer science and the Semantic Web. Web Semantics: Science, Services and Agents on the World Wide Web, 5(2), 58–71. https://doi.org/10.1016/j.websem.2007.03.002

[22] Ruohomaa, S., & Kutvonen, L. (2005). Trust management survey. Proceedings of the Third International Conference on Trust Management, 77–92. https://doi.org/10.1007/11429760_6

[23] Pirandola, S., Andersen, U. L., Banchi, L., et al. (2020). Advances in quantum cryptography. Advances in Optics and Photonics, 12(4), 1012–1236. https://doi.org/10.1364/AOP.361502

[24] Olufemi, O. D., Ikwuogu, O. F., Kamau, E., Oladejo, A. O., Adewa, A., & Oguntokun, O. (2024). Infrastructure-as-code for 5g ran, core and sbi deployment: a comprehensive review. International Journal of Science and Research Archive, 21(3), 144-167. https://doi.org/10.30574/gjeta.2024.21.3.0235

[25] Mosca, M. (2018). Cybersecurity in an era with quantum computers: Will we be ready? IEEE Security & Privacy, 16(5), 38–41. https://doi.org/10.1109/MSP.2018.3761723

[26] Chen, L. K., Jordan, S., Liu, Y. K., Moody, D., Peralta, R., Perlner, R., & Smith-Tone, D. (2016). Report on post-quantum cryptography. NISTIR 8105. https://doi.org/10.6028/NIST.IR.8105

[27] Alagic, G., Alperin-Sheriff, J., Apon, D., et al. (2020). Status report on the second round of the NIST post-quantum cryptography standardization process. NISTIR 8309. https://doi.org/10.6028/NIST.IR.8309

[28] Bernstein, D. J., & Lange, T. (2017). Post-quantum cryptography. Nature, 549(7671), 188–194. https://doi.org/10.1038/nature23461

[29] Arute, F., Arya, K., Babbush, R., et al. (2019). Quantum supremacy using a programmable superconducting processor. Nature, 574(7779), 505–510. https://doi.org/10.1038/s41586-019-1666-5

[30] Gisin, N., Ribordy, G., Tittel, W., & Zbinden, H. (2002). Quantum cryptography. Reviews of Modern Physics, 74(1), 145–195. https://doi.org/10.1103/RevModPhys.74.145

[31] Scarani, V., Bechmann-Pasquinucci, H., Cerf, N. J., et al. (2009). The security of practical quantum key distribution. Reviews of Modern Physics, 81(3), 1301–1350. https://doi.org/10.1103/RevModPhys.81.1301

[32] Bobie-Ansah, D., Olufemi, D., & Agyekum, E. K. (2024). Adopting infrastructure as code as a cloud security framework for fostering an environment of trust and openness to technological innovation among businesses: Comprehensive review. International Journal of Science & Engineering Development Research, 9(8), 168–183. http://www.ijrti.org/papers/IJRTI2408026.pdf

[33] Lo, H. K., Curty, M., & Tamaki, K. (2014). Secure quantum key distribution. Nature Photonics, 8(8), 595–604. https://doi.org/10.

[34] Aljunaid, S. K., Almheiri, S. J., Dawood, H., & Khan, M. A. (2025). Secure and Transparent Banking: Explainable AI-Driven Federated Learning Model for Financial Fraud Detection. Journal of Risk and Financial Management, 18(4), 179. https://doi.org/10.3390/jrfm18040179

[35] Balija, S. B. (2025). FedMM-X: A Trustworthy and Interpretable Framework for Federated Multi-Modal Learning in Dynamic Environments. arXiv preprint arXiv:2503.19564. https://arxiv.org/abs/2503.19564

[36] Oladejo, A. O., Adebayo, M. A., Olufemi, D., Kamau, E., Bobie-Ansah, D., & Williams, D. E. (2025). Privacy-aware ai in cloud-telecom convergence: a federated learning framework for secure data sharing. International Journal of Science and Research Archive, 15(1), 005-022. https://doi.org/10.30574/ijsra.2025.15.1.0940

[37] Dhanawat, V. (2025). AI's Role in Ethical Decision-Making: Fostering Fairness in Critical Systems with Explainable AI (XAI). IEEE Computer Society. https://www.computer.org/publications/tech-news/community-voices/explainable-ai/

[38] Olufemi, O. D., Ejiade, A. O., Ogunjimi, O., & Ikwuogu, F. O. (2024). AI-enhanced predictive maintenance systems for critical infrastructure: Cloud-native architectures approach. World Journal of Advanced Engineering Technology and Sciences, 13(02), 229–257. https://doi.org/10.30574/wjaets.2024.13.2.0552

[39] Mylrea, M., & Robinson, N. (2023). Artificial Intelligence (AI) Trust Framework and Maturity Model: Applying an Entropy Lens to Improve Security, Privacy, and Ethical AI. Entropy, 25(10), 1429. https://doi.org/10.3390/e25101429

[40] Song, H. (2024). Federated Learning for Digital Healthcare Systems. Elsevier.

[41] Ren, C., Yan, R., Zhu, H., Yu, H., Xu, M., Shen, Y., Xu, Y., Xiao, M., Dong, Z. Y., Skoglund, M., Niyato, D., & Kwek, L. C. (2023). Towards Quantum Federated Learning. arXiv preprint arXiv:2306.09912. https://arxiv.org/abs/2306.09912

[42] Zhang, Y., Zhang, C., Zhang, C., Fan, L., Zeng, B., & Yang, Q. (2022). Federated Learning with Quantum Secure Aggregation. arXiv preprint arXiv:2207.07444. https://arxiv.org/abs/2207.07444

[43] David Olufemi, Ayodeji Olutosin Ejiade, Friday Ogochukwu Ikwuogu, Phebe Eleojo Olufemi, Deligent Bobie-Ansah, 2025, Securing Software-Defined Networks (SDN) Against Emerging Cyber Threats in 5G and Future Networks – A Comprehensive Review, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 14, Issue 02 (February 2025).

[44] Pokhrel, S. R., Yang, L., Rajasegarar, S., & Li, G. (2024). Robust Zero Trust Architecture: Joint Blockchain based Federated Learning and Anomaly Detection based Framework. arXiv preprint arXiv:2406.17172. https://arxiv.org/abs/2406.17172

[45] Si-Ahmed, M., et al. (2024). Advancements in securing federated learning with IDS. Artificial Intelligence Review, 57(2), 1234–1256. https://doi.org/10.1007/s10462-024-11082-w

[46] Ogunjinmi, O. F. (2025). Optimizing network reliability: Strategies for resilient telecommunications infrastructure in emerging economies. Global Journal of Engineering and Technology Advances, 22(3), 236-258. https://doi.org/10.30574/gjeta.2025.22.3.0065

[47] Mylrea, M., & Robinson, N. (2023). Artificial Intelligence (AI) Trust Framework and Maturity Model: Applying an Entropy Lens to Improve Security, Privacy, and Ethical AI. Entropy, 25(10), 1429. https://doi.org/10.3390/e25101429

[48] Dhanawat, V. (2025). AI's Role in Ethical Decision-Making: Fostering Fairness in Critical Systems with Explainable AI (XAI). IEEE Computer Society. https://www.computer.org/publications/tech-news/community-voices/explainable-ai/