IMPLEMENTING BLOCKCHAIN FOR ENHANCED SECURITY IN FOG COMPUTING SYSTEMS: A COMPREHENSIVE REVIEW AND SIMULATION FRAMEWORK

Choon Keat Low[1*], Fung Ji Lim[2], Bee Sian Tan[3], Gar Chi Phoon[4]

Abstract

The rapid growth of Internet of Things (IoT) devices has caused a major shift from central cloud computing to decentralised "fog" computing. This new approach processes data closer to where it is created. While this change reduces delays and saves bandwidth, it also creates new security risks regarding data trust, user verification, and safe communication between different devices. This research explores adding blockchain technology to fog computing to build a secure, decentralised system. By using blockchain's permanent and open record-keeping, the proposed framework tracks where data comes from and prevents unauthorised access, even in systems that require fast responses. The study uses iFogSim, a simulation tool designed for fog environments, to test the performance of this setup. We chose iFogSim over other tools because it is better at modelling complex device layers and sensor interactions. The results show a clear trade-off: while making the blockchain harder to tamper with improves security, it also slows down the system and uses more computing power. Specifically, increasing the complexity of the security checks caused delays of nearly 30 seconds. This significantly reduced performance, especially when many devices were connected. To address these speed issues, the research explores advanced methods like "sharding" and "Layer-2 protocols" to handle more data efficiently. The findings conclude that for large IoT networks with limited resources, it is essential to use faster, lighter verification methods (such as Proof of Authority) rather than heavy, complex ones.

Received: *24 December, 2025*
Revised: *30 March, 2026*
Accepted: *25 May, 2026*
Published: *29 August, 2026*

[1,2,3]Department of Information and Communication Technology, Faculty of Computing and Information Technology, Tunku Abdul Rahman University of Management and Technology (TAR UMT), 53300 Kuala Lumpur, Malaysia.

[4]Department of Creative Industries, Faculty of Communication and Creative Industries, Tunku Abdul Rahman University of Management and Technology (TAR UMT), 53300 Kuala Lumpur, Malaysia.

****Corresponding author:***
lowck@tarc.edu.my

DOI: *https://doi.org/10.54552/bzddh935*

Keywords:
Fog Computing, Blockchain, IoT security, iFogSim, Distributed ledger technology, Sharding

1.0 INTRODUCTION

The digital world is undergoing a major change due to the widespread use of the Internet of Things. Smart devices are being connected to important systems like autonomous transport, smart grids, remote health monitoring, and factory automation. This trend creates data volumes that central computing models cannot handle easily. Experts predict that billions of devices will be connected by 2030 and will produce massive amounts of data annually. Standard cloud computing offers strong power and storage but has physical limits regarding speed and network traffic. Tasks that need immediate results, like avoiding crashes in vehicle networks or using remote surgery tools, cannot wait for data to travel to a central cloud. The delay is often too high and could lead to serious failures in these critical situations.

Fog computing has emerged as a strategic solution to these challenges. Originally defined by Cisco, this approach extends cloud functions closer to the network edge and enables local data processing, storage and networking. This decentralisation significantly reduces delay, saves backbone network bandwidth and improves location awareness. However, this architectural shift changes the security landscape. Cloud data centres typically use defined physical and digital boundaries for protection. In contrast, fog nodes are geographically dispersed and often sit in physically accessible places such as roadside units or smart building gateways. This makes them highly vulnerable to various threats, including the injection of unauthorised nodes, interception attacks and data tampering during local processing.

The primary motivation for this research lies in the urgent necessity to reconcile the performance benefits of fog computing with the rigorous security demands of modern digital infrastructure. While fog computing addresses the latency issues of the cloud, it inadvertently expands the attack surface. Traditional security mechanisms, such as centralised Certificate Authorities (CAs) or heavy asymmetric encryption protocols, often reintroduce the very latency and bottlenecks that fog computing aims to eliminate. Furthermore, centralised security models represent a single point of failure; if the central authority is compromised, the entire network is at risk.

Blockchain technology offers a robust solution to these challenges through its decentralised and transparent record system. This approach distributes the work of checking and saving transactions across a network, which removes single points of failure and builds trust. The permanent nature of the records guarantees that data origins are clear, while smart contracts automate access rules without human input. However, using blockchain in fog environments with limited resources is difficult. Standard consensus algorithms like Proof of Work require high computing power and energy. Using these heavy processes on fog nodes causes problems because

these devices often rely on batteries or have low processing power. This extra load can rapidly drain resources and slow down services, which acts against the core efficiency goals of fog computing.

Guided by these challenges, this study pursues the following objectives:

1. To comprehensively assess the security vulnerabilities inherent in decentralised fog architectures, specifically focusing on data integrity and authentication.
2. To develop a customised blockchain-based security framework simulated within the iFogSim environment that mitigates these vulnerabilities.
3. To rigorously quantify the performance trade-offs between enhanced security (via blockchain difficulty) and system efficiency (latency, throughput, and energy consumption).
4. To analyse and propose scalability solutions, such as sharding and Layer-2 protocols, to mitigate the identified computational overheads and render the architecture viable for large-scale IoT deployments.

2.0 LITERATURE REVIEW

This section provides a critical analysis of the current state of research regarding the convergence of blockchain and fog computing. It moves beyond a cursory overview to examine the architectural evolutions, security implications, consensus mechanisms, and specific technological interventions proposed in recent scholarship.

2.1.1 The Evolution: Cloud to Fog to Edge

Fog computing emerged as an extension of cloud computing, introduced by Cisco in 2012 to reduce latency and bandwidth usage by decentralising data processing closer to the network edge (Bonomi *et al.,* 2012). Initially, research focused on establishing the architecture and addressing security and privacy challenges, with recent advancements integrating technologies like edge computing and AI (Gill *et al.,* 2024). Satoshi Nakamoto introduced blockchain technology in 2008 as the foundation for Bitcoin, but the field has expanded far beyond just digital currency (Nakamoto, 2008). Ethereum added smart contracts in 2015, which allowed the technology to support decentralised finance and supply chain management (Buterin, 2015).

Current projects aim to improve capacity, security and the ability of different systems to work together. Solutions like Hyperledger Fabric now address these specific business requirements (Androulaki *et al.,* 2018).Fog computing was introduced to address the inadequacies of cloud computing in handling the "velocity, variety, and volume" of IoT data. The literature consistently identifies latency and bandwidth conservation as the primary drivers for this shift. Hazra *et al.* (2023) emphasise that while cloud computing offers superior storage, the latency induced by data transmission is unacceptable for delay-sensitive applications. However, definitions vary; "Edge Computing" typically refers to processing at the device layer itself (e.g., on the sensor or smartphone), while "Fog Computing" encompasses the continuum of infrastructure (gateways, routers, micro-datacentres) between the edge and the cloud. This distinction is crucial for security modelling, as fog nodes often possess sufficient computational capacity to perform validation tasks (mining) that edge sensors cannot.

The hierarchical nature of fog computing creates a unique environment for data flow. In a typical scenario, data moves from IoT sensors (Layer 1) to Fog Nodes (Layer 2) for immediate processing and short-term storage, and finally to the Cloud (Layer 3) for long-term analytics and archival. This multi-hop transmission path increases the risk of data modification or interception. Recent studies have highlighted that securing this "computing continuum" requires a unified trust layer that spans all three tiers, a role for which blockchain is uniquely suited.

2.1.2 Security Landscape in Fog Computing

The transition to fog computing redistributes trust. In a cloud environment, trust is centralised; in fog computing, trust must be distributed. The literature identifies three critical security pillars in this domain: Data Integrity, Secure Communication, and Access Control.

Data Integrity and Authenticity: Ensuring that data remains unchanged as it traverses multiple hops is critical. Recent frameworks suggest tracking data provenance to prevent tampering as it moves through the fog layer. Aazam *et al.* (2018) proposed a history-based verification mechanism, but such centralised approaches can face scalability limits. Blockchain's immutable ledger provides a superior alternative by creating a permanent, hash-linked record of data states at every hop. Any unauthorised modification to a record would invalidate the hash chain, making the tampering evident to all nodes in the network.

Secure Communication: The decentralised nature of fog requires dynamic secure channels. Traditional Public Key Infrastructure (PKI) can be cumbersome to manage in highly dynamic fog environments where nodes join and leave frequently. Studies have utilised blockchain to manage encryption keys, effectively replacing centralised PKI infrastructures that often act as bottlenecks. Jumani *et al.* (2023) reviewed various key management schemes and concluded that blockchain-based decentralised key distribution significantly reduces the latency associated with session establishment compared to centralised servers.

Access Control: Blockchain has been demonstrated as a utility for decentralised access control. Instead of a central server validating every request, smart contracts can automate authorisation based on immutable rules. This method, often referred to as Attribute-Based Access Control (ABAC), allows for fine-grained permissions (e.g., "User A can access Sensor B data only if Location is Hospital"). Yaraziz *et al.* (2024) demonstrated that blockchain-based access control enhances privacy and reduces verification latency by eliminating the round-trip to a central authentication server.

2.1.3 Blockchain Integration and Consensus Mechanisms

The integration of blockchain into fog computing, often termed FogChain, aims to create a trusted layer over otherwise untrusted infrastructure. However, the choice of consensus algorithm, which defines how nodes agree on the validity of transactions, is the most critical factor in determining the feasibility of this integration.

2.1.4 Proof of Work (PoW)

Traditional consensus mechanisms like Proof of Work require the system to solve difficult cryptographic puzzles. This process effectively stops attackers from creating multiple fake identities to control the network. However, the method uses excessive energy and demands high computing power. Research by Nasir *et al.* (2024) indicates that using Proof of Work on fog nodes with limited resources drains batteries rapidly. This approach also causes significant delays and makes it unsuitable for Internet of Things applications that need immediate responses.

2.1.5 Proof of Stake (PoS)

PoS addresses the energy issues of PoW by selecting validators based on their "stake" (wealth) in the network. While significantly more energy-efficient, standard PoS can lead to centralisation where the "rich get richer." In an IoT context, where devices do not hold currency, PoS must be adapted to use metrics like "reputation" or "resource contribution" as the stake.

2.1.6 Proof of Authority (PoA)

PoA is increasingly cited as the most suitable consensus mechanism for permissioned fog networks. In PoA, validation rights are restricted to a set of pre-approved, trusted nodes (authorities). This eliminates the need for mining, allowing for high transaction throughput and negligible energy consumption. Recent studies (Pajooh *et al.,* 2021) highlight PoA's ability to maintain high security in industrial environments where the identity of the fog nodes (e.g., owned by a specific factory or city municipality) is known.

2.1.7 Practical Byzantine Fault Tolerance (PBFT)

Practical Byzantine Fault Tolerance relies on a voting mechanism where nodes communicate through three distinct phases known as pre prepare, prepare and commit. This approach provides immediate confirmation and ensures the system remains functional even if one third of the nodes act maliciously. However, the amount of communication required grows rapidly as the number of nodes increases. This heavy message traffic creates a challenge for deploying the system in large fog networks that involve thousands of devices.

2.1.8 Sharding and Layer-2 Solutions

Scalability remains the "Achilles' heel" of blockchain integration in high-volume IoT networks. A linear blockchain, where every node processes every transaction, cannot match the throughput of centralised databases.

Sharding: Sharding involves partitioning the blockchain network into smaller, independent groups called "shards," each capable of processing transactions in parallel. In a fog context, sharding is often geographic (e.g., grouping nodes by physical proximity) or functional (e.g., separate shards for traffic data and environmental data). This significantly increases the transactions per second (TPS) of the system. However, maintaining security within small shards and handling cross-shard transactions (which require synchronisation between shards) introduces significant complexity.

Layer-2 Protocols: Layer 2 solutions operate above the main blockchain to move processing work away from the primary network. Methods such as Rollups collect hundreds of transactions externally and send only a single proof of validity to the main chain. State Channels allow two parties to conduct many exchanges in private and only settle the final balance on the main ledger. These strategies are vital for managing the frequent small transactions typical of smart devices. They ensure the main network does not become blocked and keep costs low.

2.2 Research Gaps Addressed

Despite the rich literature, significant gaps remain. Many studies discuss these concepts theoretically or use generic network simulators that fail to capture the specific resource constraints of fog devices. There is a lack of empirical research that:

1. Quantifies the specific energy and latency penalties of blockchain integration using a dedicated fog simulator like iFogSim.
2. Provides a direct comparison of how different difficulty levels in consensus algorithms impact the "Sense-Process-Actuate" loop of IoT applications.
3. Evaluates the practical trade-offs between security levels and system responsiveness in a simulated healthcare environment.

This research aims to bridge these gaps by providing a comprehensive simulation framework and empirical data analysis.

3.0 SYSTEM DESIGN AND METHODOLOGY

This research presents a structural framework that places blockchain features directly within the fog layer. The design focuses on the sense process actuate model common in smart device applications. It adds a verify record cycle powered by the blockchain. This double loop system ensures that data moves quickly to the control mechanisms while the system keeps a secure history log.

3.1 Justification for Using iFogSim

The selection of the simulation tool is critical for the validity of the results. iFogSim was chosen over alternatives such as CloudSim, EdgeCloudSim, or generic network simulators (like NS-3) for specific reasons derived from comparative analyses:

1. Hierarchical Modelling: Unlike CloudSim, which focuses on flattened data centre architectures and virtual machine (VM) provisioning, iFogSim natively supports the tree-like topology of Fog computing (Cloud -> Proxy -> Gateway -> End Device). This allows for the realistic modelling of data flows that traverse multiple hops up and down the hierarchy, capturing the latency introduced at each layer.
2. Sensor-Actuator Model: iFogSim includes specific Java classes for Sensor, Actuator, and Tuple, allowing for the realistic modelling of the IoT application loop. This is distinct from EdgeCloudSim, which focuses more on networking and mobility but lacks the detailed application module dependency modelling required to simulate a data processing pipeline (e.g., Sensor -> Processor -> Database).
3. Energy Models: iFogSim provides built-in classes for calculating energy consumption based on CPU usage modes. The FogLinearPowerModel class allows the simulation to track energy usage dynamically as the

workload changes, enabling the precise quantification of the "energy penalty" introduced by blockchain mining operations. This is essential for evaluating the sustainability of the proposed framework.

4. Application Placement Policies: iFogSim allows for the definition of module placement strategies (e.g., ModulePlacementEdgewards), which determine where specific application modules run. This is crucial for testing the impact of moving the "mining" module from the cloud to the fog.

3.2 Security Framework Components

The key components of the proposed framework are detailed in Table 1.

3.3 System Architecture and Methodology Flow

The system architecture consists of three logical layers: the IoT Layer (sensors generating raw data, e.g., EEG signals), the Fog Layer (intermediary nodes performing computation, mining, and local storage), and the Cloud Layer (centralised servers for long-term archival and global analytics). The interaction flow within the simulation follows a strict sequence to model a real-world transaction lifecycle. The overall methodology flow is illustrated in Figure 1.

3.4 Methodology Flow

The interaction flow within the simulation follows a strict sequence to model a real-world transaction lifecycle:

1. Data Generation: Sensors (modelled as Sensor entities in iFogSim) generate raw data tuples (e.g., EEG signals). The generation rate is determined by the transmit distribution parameter (e.g., deterministic 10ms intervals).
2. Encryption & Hashing: Before transmission, the tuple payload is processed. A SHA-256 hash of the payload is generated to create a unique fingerprint. In a full implementation, the payload would also be encrypted (e.g., AES-128), though the simulation focuses on the computational cost of hashing and consensus.
3. Fog Processing (Application Logic): The tuple is transmitted to a Fog Node (modelled as a FogDevice). The node executes the primary application logic (e.g., a ConcentrationCalculator module) to process the raw signal into a meaningful metric.
4. Blockchain Consensus (Mining): Simultaneously, the Fog Node acts as a blockchain miner. It bundles the transaction (the processed tuple and its hash) into a candidate block. The node then performs a Proof of Work (PoW) calculation based on the current Difficulty Level Bl . This involves repeatedly hashing the block header with a nonce until a value lower than the target difficulty is found. This step simulates the computational overhead of securing the block.

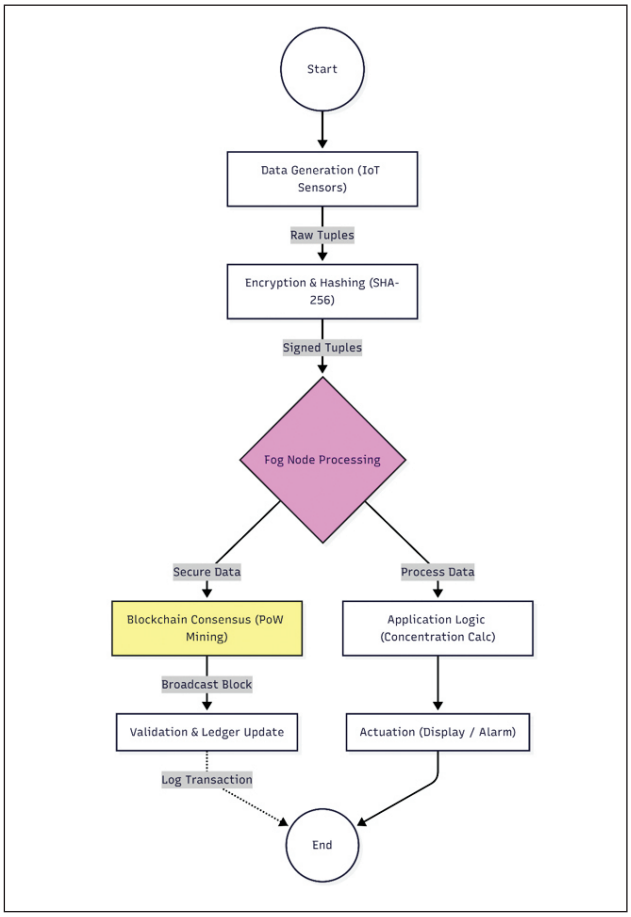

Figure 1: Methodology Flow of the Proposed Framework

Table 1: Key Security Components in the Proposed Fog Framework

Component	Functionality	Role in Framework	Technology/Mechanism
Data Integrity	Verifying accuracy and consistency of data over its lifecycle.	Ensures sensor data is not altered during transmission. Any modification to the tuple payload changes the hash, alerting the receiving node.	SHA-256 Hashing: Every data tuple generated by a sensor is hashed. The hash travels with the data and is verified at the Fog node.
Authentication	Verifying the identity of devices and users.	Fog nodes act as validators; only registered devices with valid cryptographic keys can submit transactions to the network.	Digital Signatures / PKI: Devices sign their data payloads with a private key. Fog nodes verify the signature using the device's public key stored on the blockchain.
Access Control	Restricting resource access to authorised entities.	Automates permission logic (e.g., "Allow access if UserID matches OwnerID"). Prevents unauthorised query of sensitive data.	Smart Contracts: Self-executing code stored on the blockchain defines and enforces access policies dynamically (ABAC).
Availability	Ensuring services are accessible despite failures.	Prevents single points of failure; if one fog node fails, others hold the record, ensuring data persistence and system resilience.	Distributed Ledger Technology (DLT): The ledger is replicated across multiple Fog nodes. Consensus ensures all copies remain synchronised.
Non-Repudiation	Preventing denial of validity of signed data.	Ensures devices cannot deny sending data. Provides a robust, immutable audit trail for liability and forensic analysis.	Immutable Ledger: Once a transaction (data entry) is confirmed in a block, it cannot be deleted or altered, providing cryptographic proof of origin.

Table 2: Simulation Configuration Parameters

Parameter	Value	Description/Justification
Fog Device MIPS	2800 MIPS	Processing power of the Fog Node, equivalent to a standard Raspberry Pi or mid-range IoT gateway.
Fog Device RAM	4000 MB	Memory available for application modules and the blockchain ledger (DAG storage).
Uplink Bandwidth	10 Gbit/s	High-speed connection from Fog Node to Cloud Proxy, simulating fibre backhaul.
Edge-Fog Latency	2 ms	Low latency connection representing local WiFi or 5G edge access.
Fog-Cloud Latency	100 ms	Higher latency representing the WAN connection to centralised cloud servers.
Sensor Workload	10 ms interval	Frequency of data generation by EEG sensors, representing high-frequency streaming.
Tuple CPU Length	3500 MIPS	The computational cost (instructions) required to process one data tuple.
Busy Power	107.339 W	Power consumption of the Fog Node when the CPU is fully utilised (processing + mining).
Idle Power	83.43 W	Baseline power consumption of the Fog Node when waiting for tasks.
Mining Difficulty	Variable (1-7)	The complexity parameter for the PoW algorithm, determining the required hash target.

5. Validation & Append: Once the PoW puzzle is solved, the block is broadcast to other nodes for validation. If the block is valid (correct hash, valid transactions), it is appended to the local ledger of all participating nodes.
6. Actuation: The processed application result is sent to the Actuator (e.g., a display or alarm), completing the feedback loop. Crucially, the network metrics (latency, energy consumption, network usage) are logged by iFogSim's monitoring engine throughout this entire process for analysis.

4.0 IMPLEMENTATION AND TESTING

This section details the simulation environment, specific parameters used to model the fog computing scenario, and the experimental setup.

4.1 Simulation Environment Setup

The simulation was built using the Java programming language, utilising the iFogSim toolkit libraries. The underlying hardware for the simulation host was a standard workstation (Intel Core i7 Processor, 16GB RAM, Windows 10), but the simulated environment modelled a distributed network of resource-constrained devices.

The simulation scenario models a "Smart Healthcare" application, a domain where latency is critical and data sensitivity is high. The application structure is a Directed Acyclic Graph (DAG) consisting of three modules: EEG_ Sensor (Source) -> Concentration_Calculator (Processing) -> Display_Actuator (Sink).

4.2 Simulation Parameters

To ensure reproducibility and realism, the simulation parameters were carefully selected to mirror real-world fog hardware (e.g., Raspberry Pi 3/4 gateways) and typical network conditions found in LTE/5G or WiFi environments.

4.3 Experimental Variables

The core experimental variables manipulate the blockchain's behaviour and the network density to observe their impact on the fog network's performance:

1. Difficulty Levels Bl : Tested at values of 1, 3, 5, and 7. This parameter controls the computational cost of the mining function (specifically, the number of leading zeros required in the resulting hash). Increasing this value exponentially increases the difficulty.
2. Device Scalability Dm: The number of mobile devices (sensors) connected per fog node is varied between 2, 6, and 10. This simulates low, medium, and high network congestion scenarios, allowing us to observe how the system scales under load.

4.4 Mathematical Models

The simulation relies on specific mathematical formulations to calculate key metrics.

Latency Model: The total latency (T_{total}) for a transaction is modelled as the sum of propagation delay (T_{prop}), processing time (T_{proc}), and queueing delay (T_{queue}). In our blockchain-integrated model, T_{proc} includes the time taken for the consensus algorithm($T_{consensus}$).

$$T_{total} = T_{prop} + (T_{proc\_app} + T_{consensus}) + T_{queue}$$

Energy Consumption Model: The energy consumption (E) of a fog node is calculated based on its power state over time.

$$E = \int_0^T P(t)dt \approx \sum(P_{busy} \times \Delta t_{busy}) + (P_{idle} \times \Delta t_{idle})$$

Where P_{busy} is the power consumed during processing/ mining, and P_{idle} is the base power consumption.

5.0 RESULTS AND DISCUSSION

This section presents the empirical data derived from the iFogSim simulations, interpreting the results in the context of the defined objectives. The analysis focuses on the trade-offs between blockchain security (difficulty) and system performance (latency and energy).

5.1 Average Block Creation Time

The relationship between blockchain difficulty (B_l) and the time required to mine a block is non-linear, exhibiting the exponential growth characteristic of PoW algorithms.

1. Low Difficulty (Bl = 1,3): For all device counts (Dm), The block creation time remains negligible, typically under 200 ms. This suggests that at low security settings, where the puzzle is trivial to solve, blockchain integration does not introduce significant latency penalties. The fog nodes can handle the hashing overhead without blocking the main application thread.

2. Critical Inflection Point (BI = 5): A sharp, non-linear increase in mining time is observed. For a network with 10 devices (Dm = 10), the block creation time spikes to nearly 30,000 ms (30 seconds). This indicates that the computational capacity of the fog nodes (2800 MIPS) is becoming fully saturated by the mining task. The nodes are spending more time hashing than processing sensor data.
3. High Difficulty (BI = 7): At this level, the fog nodes, which share resources between data processing and mining, become completely overwhelmed. The system effectively stalls as the mining process consumes all available CPU cycles, leading to massive queue build-ups and packet drops.

These results corroborate findings by comparative studies on lightweight consensus, confirming that standard PoW is ill-suited for resource-constrained fog environments unless the difficulty is kept artificially low. However, keeping difficulty low compromises security, making the chain vulnerable to brute-force attacks by adversaries with even moderate computing power.

5.2 Impact on Execution Latency (Performance Trade-Off)

The simulation also measured the Average Execution Time of the primary application tasks (calculating concentration from EEG signals).

1. Resource Contention: The results show that as blockchain difficulty increases, the execution time for non-blockchain tasks also increases significantly. For instance, at Dm = 10 and BI = 6, execution time jumps to ~190 ms, compared to ~150 ms at lower difficulties.
2. Implication: This demonstrates a critical side-effect: Resource Contention. Since the Fog Node's CPU is shared, the intensive hashing required for mining "steals" CPU cycles from the primary application logic. This confirms that security mechanisms in fog computing cannot be viewed in isolation; they directly impact the Quality of Service (QoS) of the application. High-security settings (high difficulty) degrade the real-time performance of the health monitoring system, potentially delaying critical alerts.

5.3 Energy Consumption Analysis

Energy consumption is a vital metric for fog nodes, which may be battery-powered or energy-constrained. The simulation tracked the total energy consumed by the fog layer.

1. Energy Overhead: At low difficulty levels BI = 1, the energy consumption is dominated by the application logic and data transmission. However, as difficulty increases to BI = 5, the energy consumption spikes disproportionately. The nodes spend extended periods in the BUSY power state (107W) attempting to solve the PoW puzzle.
2. Sustainability Concern: The results indicate that implementing high-difficulty PoW in a fog network would drastically increase the operational carbon footprint and reduce the lifespan of battery-operated nodes. This reinforces the necessity for energy-efficient consensus mechanisms.

5.4 Discussion: Scalability and Solutions

The simulation results underscore a fundamental tension: maintaining high security (via high-difficulty PoW) renders the system unusable for real-time fog applications due to latency and energy costs. To address the scalability concerns, we propose the following solutions based on the findings and broader literature:

1. Sharding for Parallel Processing: To address the bottlenecks identified in Section 5.1, Sharding offers a viable path forward. Sharding involves partitioning the blockchain network into smaller, manageable groups or "shards," each capable of processing transactions in parallel. In a fog context, nodes can be grouped by geography (e.g., "Hospital Ward A" shard, "Traffic Intersection B" shard). This strategy reduces the validation workload on individual nodes and increases the overall speed of data processing. A node only checks transactions within its own group rather than validating the entire network activity. Communication protocols between different shards then ensure global consistency. Simulation studies indicate that sharding increases performance significantly.
2. Layer-2 Protocols (Rollups): Instead of writing every sensor reading to the main blockchain, fog nodes can aggregate thousands of readings off-chain using Rollups. In this model, the fog nodes perform the computation and state updates locally (Layer 2) and only submit a cryptographic proof (validity proof or fraud proof) to the main chain (Layer 1) periodically (e.g., every 10 minutes). This drastically reduces the "Average Block Creation Time" and network bandwidth usage seen in our results, as the main chain is not clogged with high-frequency micro-transactions. This effectively decouples the sensor data rate from the blockchain's block time.
3. Transition to Lightweight Consensus: The study explicitly confirms that PoW is suboptimal for Fog. Future implementations should transition to Proof of Authority (PoA) or Practical Byzantine Fault Tolerance (PBFT).
    - PoA: In PoA, identity serves as the stake; known and reputable fog nodes (e.g., those owned by the hospital) are given the authority to validate blocks. This eliminates the need for mining, reducing the block creation time to near-zero and removing the energy penalty.
    - PBFT: PBFT provides a voting-based consensus that is robust and fast for smaller, permissioned networks typical of fog deployments. It offers instant finality, which is crucial for real-time applications.

6.0 CONCLUSION

This study presents a comprehensive evaluation of integrating blockchain technology to enhance security within fog computing systems. Using the iFogSim simulation environment, we successfully modelled the interaction between sensors, fog nodes, and the blockchain layer to quantify performance trade-offs. The research identified critical vulnerabilities inherent in decentralised fog architectures, specifically regarding susceptibility to tampering and unauthorised access.

Furthermore, the simulation demonstrated the feasibility of injecting security modules directly into the standard data flow of the fog environment. The empirical results indicate a significant non-linear relationship between blockchain difficulty and system latency. Increasing the difficulty parameter from

1 to 5 resulted in an exponential rise in latency, reaching nearly 30 seconds, which renders high-difficulty Proof-of-Work (PoW) mechanisms unsuitable for real-time IoT applications. Additionally, the data established that PoW-based security imposes a severe energy penalty on resource-constrained fog nodes. While the integration of blockchain provides necessary guarantees regarding data immutability and provenance, the computational overhead of PoW creates a bottleneck that significantly degrades application performance.

Consequently, for fog environments to maintain both security and efficiency, implementation strategies must transition towards lightweight consensus algorithms, such as Proof of Authority (PoA) or Practical Byzantine Fault Tolerance (PBFT), or employ sharding techniques rather than traditional PoW. The computational cost associated with PoW exceeds the energy and latency budgets typical of fog infrastructure. Future research will expand the simulation framework to evaluate partitioned network designs and Layer 2 solutions. Specifically, we aim to develop a dynamic grouping algorithm that adjusts shard sizes based on real-time network traffic. Furthermore, the project plans to incorporate artificial intelligence to detect anomalies within fog nodes, identifying malicious actors prior to the consensus process. ■

ACKNOWLEDGEMENT

The authors would like to acknowledge Tunku Abdul Rahman University of Management and Technology (TAR UMT), Malaysia, for supporting this study.

AUTHORS' CONTRIBUTIONS

Choon Keat Low	Conceptualisation, Writing—original draft preparation and literature review, study design, data collection, methodology, software and data analyses
Fung Ji Lim	Data validation, visualisation, supervision and formal analysis
Bee Sian Tan	Data validation, visualisation, supervision and formal analysis
Gar Chi Phoon	Data validation, visualisation, and software implementation

REFERENCES

[1] Aazam, M., Zeadally, S., & Harras, K. A. (2018). Fog computing architecture, evaluation, and future research directions. *IEEE Communications Magazine, 56*(5), 46-52. https://doi.org/10.1109/MCOM.2018.1700707

[2] Alzoubi, Y. I., Al-Ahmad, A., Kahtan, H., & Jaradat, A. (2022). Internet of Things and blockchain integration: Security, privacy, technical, and design challenges. *Future Internet, 14*(7), Article 216. https://doi.org/10.3390/fi14070216

[3] Androulaki, E., Barger, A., Bortnikov, V., Cachin, C., Christidis, K., De Caro, A., Enyeart, D., Ferris, C., Laventman, G., Manevich, Y., Muralidharan, S., Murthy, C., Nguyen, B., Sethi, M., Singh, G., Smith, K., Sorniotti, A., Stathakopoulou, C., Vukolić, M., & Yellick, J. (2018). Hyperledger Fabric: A distributed operating system for permissioned blockchains. In *Proceedings of the Thirteenth EuroSys Conference* (pp. 1-15). ACM. https://doi.org/10.1145/3190508.3190538

[4] Atlam, H., Walters, R., & Wills, G. (2018). Fog computing and the Internet of Things: A review. *Big Data and Cognitive Computing, 2*(2), Article 10. https://doi.org/10.3390/bdcc2020010

[5] Baker, S. A., Rashid, S. J., & Alsaif, O. I. (2023). Fog computing: A comprehensive review of architectures, applications, and security challenges. *NTU Journal of Engineering and Technology, 2*(2). https://doi.org/10.56286/ntujet.v2i2.614

[6] Baniata, H., & Kertesz, A. (2020). PF-BVM: A privacy-aware fog-enhanced blockchain validation mechanism. In *Proceedings of the 10th International Conference on Cloud Computing and Services Science* (pp. 430-439). SciTePress. https://doi.org/10.5220/0009487904300439

[7] Bonomi, F., Milito, R., Zhu, J., & Addepalli, S. (2012). Fog computing and its role in the Internet of Things. In *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing* (pp. 13-16). ACM. https://doi.org/10.1145/2342509.2342513

[8] Buterin, V. (2015). *Ethereum white paper.* https://ethereum.org/en/whitepaper/

[9] Chen, Y., & Bellavitis, C. (2019). Decentralized finance: Blockchain technology and the quest for an open financial system. *SSRN Electronic Journal.* https://doi.org/10.2139/ssrn.3418557

[10] Das, R., & Inuwa, M. M. (2023). A review on fog computing: Issues, characteristics, challenges, and potential applications. *Telematics and Informatics Reports, 10*, Article 100049. https://doi.org/10.1016/j.teler.2023.100049

[11] Deepak, S., Kaur, K., Kumar, N., Alazab, M., Guizani, M., & Hossain, M. S. (2024). Exploring the potential of blockchain technology in an IoT-enabled environment: A review. *IEEE Access, 12*, 31197-31227. https://doi.org/10.1109/ACCESS.2024.3366656

[12] Gill, S. S., Arya, R. C., Wander, G. S., & Buyya, R. (2018). Fog-based smart healthcare as a big data and cloud service for heart patients using IoT. In *Lecture Notes on Data Engineering and Communications Technologies* (pp. 1376-1383). Springer. https://doi.org/10.1007/978-3-030-03146-6_161

[13] Gupta, H., Dastjerdi, A. V., Ghosh, S. K., & Buyya, R. (2017). iFogSim: A toolkit for modeling and simulation of resource management techniques in the Internet of Things, edge and fog computing environments. *Software: Practice and Experience, 47*(9), 1275-1296. https://doi.org/10.1002/spe.2509

[14] Hazra, A., Rana, P., Adhikari, M., & Amgoth, T. (2023). Fog computing for next-generation Internet of Things: Fundamental, state-of-the-art and research challenges. *Computer Science Review, 48,* Article 100549. https://doi.org/10.1016/j.cosrev.2023.100549

[15] Jumani, A. K., Shi, J., Laghari, A. A., Hu, Z., Nabi, A. U., & Qian, H. (2023). Fog computing security: A review. *Security and Privacy, 6*(6). https://doi.org/10.1002/spy2.313

[16] Kiwelekar, A. W., Patil, P., Netak, L. D., & Waikar, S. U. (2021). Blockchain-based security services for fog computing. In *Advances in Information Security* (pp. 271-290). Springer. https://doi.org/10.1007/978-3-030-57328-7_11

[17] Lawton, G. (2024, January 12). *Top 9 blockchain platforms to consider in 2024*. SearchCIO. https://www.techtarget.com/searchcio/feature/Top-9-blockchain-platforms-to-consider

[18] Nadeem, M. A., & Saeed, M. A. (2016). Fog computing: An emerging paradigm. In *2016 Sixth International Conference on Innovative Computing Technology (INTECH)* (pp. 83-86). IEEE. https://doi.org/10.1109/INTECH.2016.7845043

[19] Mukherjee, M., Matam, R., Shu, L., Maglaras, L., Ferrag, M. A., Choo, K.-K. R., & Wang, C. (2017). Security and privacy in fog computing: Challenges. *IEEE Access, 5,* 19293-19304. https://doi.org/10.1109/ACCESS.2017.2749422

[20] Mohammed Alsamman, Y., Fazea, Y., Mohammed, F., & Kehail, M. A. M. (2023). Fog computing in smart cities: A systematic review. *IEEE Conference Proceedings.* https://ieeexplore.ieee.org/document/10293505

[21] Nasir, N. M., Hassan, S., & Mohd Zaini, K. (2024). Securing permissioned blockchain-based systems: An analysis on the significance of consensus mechanisms. *IEEE Access, 12,* 138211-138238. https://doi.org/10.1109/ACCESS.2024.3465869

[22] Varshney, P., & Simmhan, Y. (2017). Demystifying fog computing: Characterizing architectures, applications and abstractions. In *2017 IEEE First International Conference on Fog and Edge Computing (ICFEC)* (pp. 115-124). IEEE. https://doi.org/10.1109/ICFEC.2017.20

[23] Pajooh, H. H., Rashid, M., Alam, F., & Demidenko, S. (2021). Hyperledger Fabric blockchain for securing the edge Internet of Things. *Sensors, 21*(2), Article 359. https://doi.org/10.3390/s21020359

[24] Ren, K., Ho, N., Loghin, D., Nguyen, T., Ooi, B. C., Ta, Q., & Zhu, F. (2023). Interoperability in blockchain: A survey. *IEEE Transactions on Knowledge and Data Engineering, 35*(12), 12750-12769. https://doi.org/10.1109/TKDE.2023.3275220

[25] Shakila, N. M., Pandiaraj, N. D. S., Sharmila, N. S. L., K., N. K. T. R., Ramalingam, N. V., & Prakash, N. D. M. (2024). Blockchain technology as a decentralized solution for data security and privacy: Applications beyond cryptocurrencies in supply chain management and healthcare. *Nanotechnology Perceptions, 20*(S14), 793-805.

[26] Stojmenovic, I., & Wen, S. (2014). The fog computing paradigm: Scenarios and security issues. *Annals of Computer Science and Information Systems.* https://doi.org/10.15439/2014F503

[27] Vashisht, P., Bajaj, S. B., & Narang, A. (2024). Energy-efficient fog computing: A review and future directions. *International Journal of Innovative Research in Computer Science and Technology, 12*(2), 135-139. https://doi.org/10.55524/ijircst.2024.12.2.24

[28] Wood, G. (2014). *Ethereum: A secure decentralized generalized transaction ledger* (Ethereum Yellow Paper). https://ethereum.github.io/yellowpaper/paper.pdf

[29] Nakamoto, S. (2008). *Bitcoin: A peer-to-peer electronic cash system.* https://bitcoin.org/bitcoin.pdf

[30] Yaraziz, M. S., Safa, N. S., & Azad, M. A. (2024). Edge computing in IoT for smart healthcare. *Journal of Ambient Intelligence and Smart Environments.* Advance online publication. https://doi.org/10.3233/AIS-230009

[31] Yi, S., Hao, Z., Qin, Z., & Li, Q. (2015). Fog computing: Platform and applications. In 2015 *Third IEEE Workshop on Hot Topics in Web Systems and Technologies (HotWeb)* (pp. 73-78). IEEE. https://doi.org/10.1109/HotWeb.2015.22

PROFILES

CHOON KEAT LOW is a lecturer at the Department of Information and Communication Technology, Tunku Abdul Rahman University of Management and Technology (TAR UMT), Malaysia. He has published extensively in reputable journals such as the International Journal of Web Services Research, contributing significantly to the fields of Cloud Computing, Fog Computing, and the Internet of Things (IoT). His research emphasises QoE-aware application mapping, energy optimisation, and deep reinforcement learning.
Email address: lowck@tarc.edu.my

FUNG JI LIM is an Assistant Professor and Programme Leader at the Faculty of Computing and Information Technology (FOCS), TAR UMT. He is a Professional Technologist (Ts.) with a strong background in software engineering and database technologies. His research expertise encompasses software quality assurance, robotic process automation, and database migration strategies for NoSQL systems, with a focus on improving data efficiency and system evolution.
Email address: limfj@tarc.edu.my

BEE SIAN TAN is an Assistant Professor at the Department of Information and Communication Technology, Faculty of Computing and Information Technology (FOCS), TAR UMT. Her research portfolio spans game-based learning, soft skills development, and the application of Augmented Reality (AR) in training environments. She is dedicated to exploring innovative pedagogical approaches and interactive technologies to enhance educational outcomes.
Email address: tanbs@tarc.edu.my

GAR CHI PHOON is an Assistant Professor at the Department of Creative Industries, Faculty of Communication and Creative Industries (FCCI), TAR UMT. Her research interests lie in digital creative multimedia, including 2D and 3D animation and Virtual Reality (VR) technologies for education. She focuses on the intersection of creative arts and technology to develop immersive learning experiences.
Email address: phoongc@tarc.edu.my