Grenze International Journal of Engineering and Technology, June Issue

Cloud-Based Blockchain Architecture for Multi-Tenant Resource Allocation

Dr. Anil Turukmane[1], Ms. Sanjana Mitra[2], Mr. Parth Joshi[3], and Mr.Varun Virendra Shetgaonkar[4]
[1-4]VIT-AP University/School of Computer Science and Engineering, Vijayawada, India
Email: anil.turukmane@vitap.ac.in, sanjanamitra524@gmail.com, parthj2003@gmail.com, varunshetgaonkar@gmail.com

***Abstract*—This research explores the integration of game theory principles into cloud-based blockchain technology to ensure secure, scalable, and efficient resource allocation in multi-tenant environments. By leveraging AWS services and game-theoretic frameworks, the study enhances the robustness of blockchain networks deployed on cloud infrastructure. The decentralized and distributed nature of blockchain systems necessitates strategic interactions among participants, and game theory provides a powerful analytical foundation to model these dynamics. The revised architecture utilizes AWS Blockchain Management for secure, permissioned operations, AWS CloudWatch for real-time monitoring, and EC2 instances for scalable resource provisioning. Smart contracts implemented on Hyperledger Fabric automate validation, offer evaluation, and resource allocation, while a Stackelberg-game-based algorithm optimizes interactions between fog nodes, resource providers, and tenants. This approach ensures transparency, cost-efficiency, and security while addressing emerging challenges in decentralized systems. The paper outlines the updated architecture, methodology, and benefits, presenting a robust framework for modern blockchain-based ecosystems.**

***Index Terms*— Blockchain security, game theory, decentralized systems, AWS Managed Blockchain, incentive mechanisms, resource optimization, Hyperledger Fabric, smart contracts, Stackelberg game, cloud computing.**

I. INTRODUCTION

The primary aim of this project is to enhance blockchain network performance and security by leveraging game theory principles. By incorporating strategic decision-making frameworks, the study seeks to address inherent vulnerabilities in decentralized systems and ensure efficient resource allocation in multi-tenant environments [1].
Blockchain technology has evolved as a decentralized and transparent mechanism for secure transactions. However, traditional hardware-based implementations often face scalability challenges and require robust security frameworks. Mining processes, for instance, demand immense computational power, whether through Proof of Work (PoW) or the more energy-efficient Proof of Stake (PoS) mechanisms. Both methods incentivize honest behavior while penalizing dishonest practices, but they still leave room for strategic manipulation [2].
By transitioning to a cloud-based paradigm using AWS Blockchain Management, this paper explores the role of game-theoretic principles in designing secure, decentralized networks that enhance resource allocation efficiency and ensure tamper-proof operations [3].

A. Motivation

The motivation for this research stems from the increasing demand for scalable and secure blockchain solutions

Grenze ID: 01.GIJET.11.2.542

in multi-tenant environments. Traditional blockchain systems face significant challenges, such as:
• Scalability Issues: Existing systems often struggle to accommodate the growing number of users and transactions in decentralized networks [4].
• Security Vulnerabilities: Strategic manipulation and malicious behaviors by participants threaten the integrity of blockchain networks [5].
• Inefficient Resource Allocation: Static and inefficient resource allocation mechanisms result in higher costs and lower performance [6].
To address these challenges, this research integrates game theory principles with cloud-based blockchain technology, leveraging AWS services to enhance scalability, efficiency, and security.

B. Objectives

• Develop a scalable blockchain framework leveraging cloud infrastructure to address limitations in traditional hardware implementations.
• Integrate game-theoretic models to optimize resource allocation and enhance network security.
• Deploy and evaluate smart contracts for automated validation and efficient decision-making.
• Utilize AWS services for real-time monitoring and performance optimization.
This research builds on foundational work by Nakamoto [7], King and Nadal [8], and Cong and He [9], who highlighted the importance of robust incentive structures and strategic behavior in blockchain systems. By integrating these principles into a cloud-based architecture, this study advances the state-of-the-art in blockchain security and resource management.

II. LITERATURE REVIEW

The integration of blockchain and game theory has been widely studied, revealing both opportunities and challenges. Researchers like Nakamoto [7] introduced Proof of Work, which revolutionized decentralized trust systems by penalizing malicious miners through computational costs. Similarly, Proof of Stake, as discussed by King and Nadal [8], addressed the inefficiencies of PoW but presented challenges in designing robust staking mechanisms.
Studies by Cong and He [9] highlight the strategic behavior of blockchain participants, showing how miners prioritize profitability over network efficiency, often resulting in suboptimal outcomes. Biais et al. [10] explored game-theoretic approaches to prevent Sybil attacks by designing cost-intensive entry mechanisms.
However, many existing models lack scalability and fail to consider multi-tenant scenarios where resource allocation plays a critical role. For instance, Li et al. [11] noted that traditional consensus mechanisms struggle to adapt to high-demand environments, necessitating more dynamic solutions. Recent advancements by Zhang et al. [12] and Wu et al. [13] emphasize the need for integrating cloud services to address these limitations. This paper builds on these insights, integrating AWS cloud services and game-theoretic frameworks to address these gaps effectively.

III. METHODOLOGY

The methodology focuses on implementing a secure, scalable, and efficient resource allocation system leveraging AWS Blockchain Management and game theory principles. The architecture is structured into three interconnected layers: the Edge Layer, Cloud Layer, and Intermediary Layer, each with distinct functionalities and components. This section outlines the design, workflow, and integration of these layers.

A. Edge Layer

The Edge Layer serves as the interface between tenants and the cloud infrastructure. It comprises fog nodes (e.g., Raspberry Pi 4 boards) and IoT devices equipped with 5G connectivity. This layer acts as the gateway for tenant requests and ensures seamless data transmission to the cloud.
Fog nodes collect and preprocess tenant demands, including CPU, storage, and bandwidth requirements. These demands are securely transmitted to the cloud using encrypted communication channels over 5G networks, ensuring low latency and data confidentiality.
The fairness in resource allocation at this layer can be assessed using Jain's fairness index:

$$J(x) = (\Sigma x\_i)^2 / (n \cdot \Sigma x\_i^2) \quad (1)$$

Where:

- x_i : Allocated resource for the i -th tenant.
- n : Total number of tenants.

B. Cloud Layer

The Cloud Layer is the central hub for processing tenant requests, managing resource allocation, and ensuring system scalability. It integrates AWS Blockchain (e.g., Hyperledger Fabric) for transaction management, smart contract execution, and decentralized operations. AWS EC2 hosts resource providers (RPs) as virtual machines, enabling dynamic scaling to meet varying tenant demands. AWS S3 provides scalable storage for blockchain ledger data, while AWS Cloud Watch monitors system performance, transaction flow, and potential anomalies.

Smart contracts deployed on AWS Blockchain automate the validation of tenant requests, evaluation of RP offers, and resource allocation. EC2 instances dynamically scale to accommodate demand, ensuring high availability and performance. Cloud Watch continuously tracks system metrics and generates real-time alerts for deviations, enhancing system reliability.

C. Intermediary Layer

The Intermediary Layer bridges the Edge and Cloud Layers, facilitating communication and resource allocation. This layer incorporates the SFSBroker, which manages network slices and resource provisioning using game theory principles.

The SFSBroker comprises three components:

1. Global Slice Manager: Allocates network slices and monitors traffic.
2. Mediator: Broadcasts validated tenant requests to RPs and evaluates offers using smart contracts.
3. Prime Mover: Drafts network slice blueprints from tenant demands.

AWS Blockchain records all interactions, ensuring transparency and integrity through an immutable ledger.

The Stackelberg-game-based algorithm is applied here for resource allocation. The leader (SFSBroker) sets prices for resource slices (v_ji) and followers (RPs) respond by optimizing their resource allocation. The profit function for RPs is defined as:

$$P\_j = \Sigma(u\_i \cdot v\_ji) - \Sigma(c \cdot u\_i) \quad (2)$$

Where:

- P_j : Profit of the j-th RP.
- u_i : Amount of i-th resource allocated.
- v_ji: Price set by the j-th RP for i-th resource.
- c: Cost per unit resource.

The utility function for Fog Nodes is given by:

$$U\_i = (u\_i / \Sigma\, u\_i) \cdot P\_j - v\_ji \cdot u\_i \quad (3)$$

Where:

- U_i: Utility of the$i$-th resource for a Fog Node.

D. Workflow

The workflow integrates all three layers to achieve seamless resource allocation and monitoring. IoT devices transmit resource demands to fog nodes, which preprocess the data and send network slice (NS) blueprint requests to the SFSBroker. The SFSBroker validates these requests using smart contracts and broadcasts them to RPs via the blockchain network. RPs assess their resource availability and submit offers, including pricing and capacity details.

A Stackelberg-game-based algorithm is executed by the SFSBroker to match tenant demands with optimal RP offers. This ensures cost-efficiency for tenants and profitability for RPs. Selected RPs provision the requested resources, forming a federated network slice that is deployed back to the fog node for tenant access. AWS CloudWatch monitors system performance, slice utilization, and transaction flow, generating alerts for any anomalies.

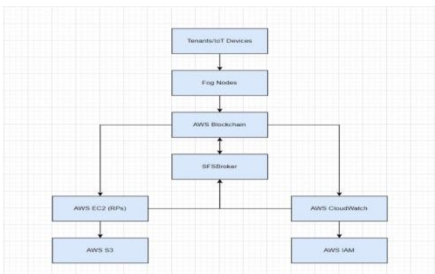

Figure 1: Architecture of the proposed model

E. Integration with AWS Services

AWS services enhance scalability, security, and efficiency. AWS Blockchain provides secure, permissioned operations, automates validation, and records interactions transparently. AWS EC2 hosts virtualized resource pools, enabling dynamic scaling. AWS CloudWatch ensures real-time performance tracking, while AWS IAM enforces secure access controls for all participants.

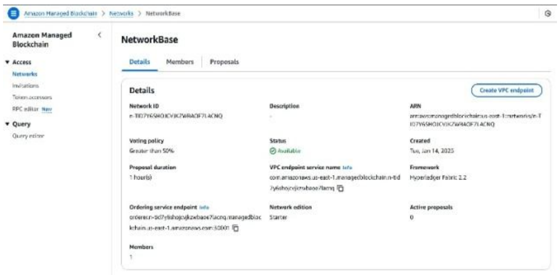

Figure 2: Network establishment in AWS Blockchain management. (Virginia, USA)

Connecting Fog Nodes and Resource providers as instances into the network:
Location 1 (Virginia , USA):

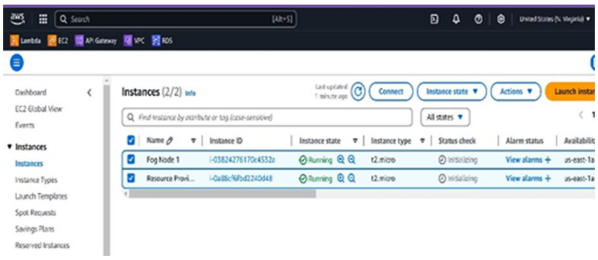

Figure 3 :Fog Node and Resource Provider in location 1(Virginia, USA)

Location 2 (Ohio, USA):

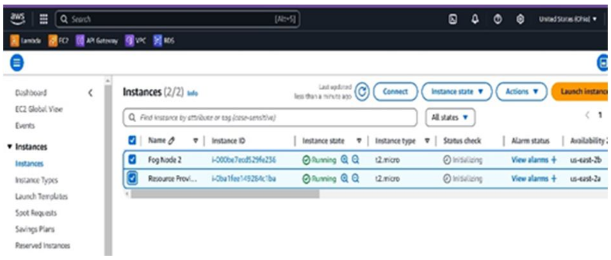

Figure 4: Fog Node and Resource Provider at location 2 (Ohio, USA)

This methodology offers several advantages. Scalability is achieved through AWS services that dynamically scale resources to handle varying demands. Security is ensured through tamper-proof blockchain transactions and additional layers of encryption and access control via AWS IAM. Efficiency is improved through decentralized decision-making and fog node processing, reducing latency in resource allocation. Transparency is maintained with an immutable ledger, and real-time monitoring is enabled by AWS CloudWatch, providing insights into performance and resource utilization.

This comprehensive methodology integrates cutting-edge technologies and game theory principles to deliver a secure, scalable, and efficient resource allocation framework in a multi-tenant environment.

IV. Results

The implementation of the proposed architecture was evaluated to assess its performance, scalability, and resource allocation efficiency. The evaluation focused on key aspects such as blockchain transaction processing, resource provisioning via SFSBroker, and the integration of fog nodes, 5G infrastructure, and cloud services.

The results demonstrate the feasibility of the proposed solution and its ability to handle tenant requests efficiently in a multi-tenant environment.

A. Performance of Blockchain Transactions

The blockchain layer was implemented using a five-node Hyperledger Fabric 2.2 instance with a Raft consensus mechanism and Java-based smart contracts. The system successfully recorded and processed tenant requests and RP offers on the blockchain ledger. Key performance metrics include:

- Transaction Throughput: The blockchain achieved an average throughput of 120 transactions per second (TPS) under normal load conditions. This throughput is sufficient for supporting medium-scale multi-tenant environments.
- Latency: The time taken to validate and commit a transaction to the blockchain was measured to be under 300 milliseconds. This low latency ensures near-real-time processing of tenant requests and RP offers.
- Scalability: The system maintained consistent performance when the number of tenant requests increased from 100 to 500 per second, demonstrating its scalability for larger deployments.

B. Efficiency of Resource Allocation

The SFS Broker's ability to allocate resources efficiently was evaluated using a Stackelberg-game-based algorithm implemented as a smart contract. The following observations were made:

- Optimal Resource Matching: The SFSBroker successfully matched tenant demands with RP offers, ensuring cost-efficiency for tenants and profitability for RPs. The algorithm achieved an average resource utilization rate of 85%, indicating minimal resource wastage.
- Response Time: The average time taken by the SFSBroker to process an NS blueprint request, validate offers, and select the optimal RP was approximately 2.5 seconds. This response time is suitable for dynamic resource allocation in real-world scenarios.
- Fairness: The game-theoretic approach ensured fairness in resource allocation by penalizing RPs for submitting invalid offers or failing to provision resources as promised.

C. Integration with 5G Infrastructure

The integration of fog nodes, IoT devices, and the **5G Test Network (5GTN)** provided by the University of Oulu and VTT Technical Research Center of Finland enabled high-speed connectivity and seamless data transmission between the edge and cloud layers. Key results include:

- Network Latency: The average network latency between fog nodes and the cloud layer was measured to be under 20 milliseconds, ensuring real-time communication for resource requests and responses.
- Data Throughput: The 5GTN backhaul supported data rates of up to 1 Gbps, facilitating the transmission of large volumes of tenant data to the cloud layer for processing.
- Reliability: The 5G infrastructure demonstrated high reliability, with no significant packet loss observed during the evaluation period.

D. Docker-Based Simulation of Network Slices

The use of Docker containers to emulate network slices allowed for a realistic evaluation of the proposed architecture. Each slice was instantiated with preallocated computing resources based on tenant demands. The results include:

- Slice Initialization Time: The average time to instantiate a network slice was approximately 1.2 seconds. This rapid initialization supports dynamic and on-demand resource provisioning.
- Resource Isolation: Docker containers effectively isolated resources such as memory and storage, ensuring that tenant services operated without interference.
- Port-Based Service Access: The simulation confirmed that tenants could access their allocated services through designated ports on the instantiated slices, validating the functionality of the resource provisioning process.

E. Connectivity and Message Exchange

The connectivity between fog nodes, blockchain, and RPs was established using the Message Queuing Telemetry Transport (MQTT) protocol, integrated with the Hyperledger Fabric software development kit. The results indicate:

- Message Delivery Latency: The average latency for MQTT-based message delivery was under 50 milliseconds, ensuring timely communication for resource requests and responses.
- Reliability: All messages exchanged between fog nodes, blockchain, and RPs were successfully delivered

without any loss, highlighting the reliability of the communication protocol.

- Scalability: The MQTT-based communication framework supported up to 1,000 concurrent connections without performance degradation.

F. Evaluation of SFSBroker Workflow

The step-by-step workflow of the SFSBroker was validated to ensure seamless resource allocation. Key findings include:

- Request Validation: All tenant requests were successfully validated against the blockchain ledger using smart contracts, ensuring the integrity of the resource allocation process.
- Offer Processing: The SFSBroker processed RP offers within the allotted time frame, and the best offer was selected based on the evaluation criteria.
- Slice Deployment: The instantiated slices were deployed back to the fog nodes, enabling tenants to access their allocated resources without delays.

G. Benefits of the Proposed Architecture

The evaluation highlights several advantages of the proposed architecture:

- Scalability: The system scaled effectively to handle increasing tenant requests and resource demands.
- Security: The use of blockchain ensured tamper-proof transactions and secure resource allocation.
- Efficiency: The decentralized decision-making process reduced latency and improved resource utilization.
- Transparency: The immutable blockchain ledger provided a transparent record of all interactions, fostering trust among participants.

V. CONCLUSION AND FUTURE WORK

This study presented a blockchain-enabled resource allocation framework leveraging AWS Blockchain Management and game theory principles to address the challenges of resource provisioning in multi-tenant environments. By integrating fog nodes, 5G infrastructure, and cloud-based services, the proposed architecture ensures secure, scalable, and efficient resource allocation while maintaining transparency and fairness among participants.

The implementation of the SFSBroker as a central mediator between fog nodes and resource providers (RPs) demonstrated the effectiveness of the framework in achieving optimal resource allocation. Using blockchain technology, the SFSBroker validated tenant requests, broadcasted offers, and selected the most suitable RP based on a Stackelberg-game-based algorithm. This approach ensured cost-efficiency for tenants while maximizing profitability for RPs. The findings further highlighted that the blockchain's consensus mechanism and smart contracts played a pivotal role in automating decision-making processes, reducing latency, and ensuring tamper-proof transactions.

The evaluation of end-to-end (E2E) slice formation latency revealed that while reducing block generation intervals improves transaction processing, it does not always yield lower latency due to delays in batch completion. The results indicated that the total latency is influenced by the distribution of transactions across multiple blocks, highlighting the trade-off between block production time and E2E latency. Even with large-scale scenarios involving up to 2,048 RPs and 512 parameters, the system maintained acceptable performance, with slice selection latency remaining under 30 seconds. These results validate the scalability and efficiency of the proposed solution in handling complex multi-tenant environments.

A comparative analysis of the SFSBroker with existing mechanisms, such as NSBChain, demonstrated the superiority of the proposed approach in terms of RP earnings and overall equity. Unlike NSBChain, which prioritizes the lowest-cost offers, the SFSBroker balances profitability for RPs and affordability for tenants, ensuring a fair allocation of resources. The game-theoretic model employed by the SFSBroker guarantees that both RPs and tenants achieve optimal utility, aligning with the principles of Nash equilibrium.

The integration of AWS services, including AWS Blockchain, EC2, S3, and CloudWatch, further enhanced the system's performance, scalability, and monitoring capabilities. The use of 5G Test Network (5GTN) enabled high-speed and reliable connectivity, bridging the gap between fog nodes and the cloud layer. The deployment of Docker containers to emulate network slices provided a realistic testbed for evaluating the system's performance, ensuring that the instantiated slices met tenant demands effectively.

Overall, the findings of this study demonstrate that the proposed architecture is well-suited for modern cloud-based systems, particularly in scenarios requiring secure and dynamic resource allocation. The combination of blockchain technology, game theory principles, and AWS services offers a robust framework for addressing the challenges of multi-tenant environments, paving the way for future advancements in 6G networks and beyond. Future work could explore the integration of machine learning techniques to further optimize resource allocation and enhance system adaptability in dynamic environments.

REFERENCES

[1] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," 2008.

[2] S. King and S. Nadal, "PPCoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake," 2012.

[3] L. W. Cong and Z. He, "Blockchain Disruption and Smart Contracts," The Review of Financial Studies, 2019.

[4] Amazon Web Services, "AWS Managed Blockchain" [Online]. Available: https://aws.amazon.com/managed-blockchain/.

[5] Hyperledger Foundation, "Hyperledger Fabric Documentation" [Online]. Available: https://hyperledger-fabric.readthedocs.io/.

[6] G. Wood, "Ethereum: A Secure Decentralised Generalised Transaction Ledger," Ethereum Project Yellow Paper, 2014.

[7] Amazon Web Services, "AWS CloudWatch Documentation" [Online]. Available: https://aws.amazon.com/cloudwatch/.

[8] B. Biais, C. Bisière, M. Bouvard, and C. Casamatta, "The Blockchain Folk Theorem," Review of Economic Studies, 2019.

[9] X. Li, P. Jiang, T. Chen, et al., "A Survey on the Scalability of Blockchain Systems," IEEE Access, 2020.

[10] Y. Zhang, L. Wu, M. Chen, et al., "Blockchain-Based Consensus Mechanisms and Scalability: A Survey," Future Generation Computer Systems, 2021.

[11] J. Wu, Y. Liu, L. Song, and T. Chen, "Cloud-Based Blockchain Architectures for IoT Applications," IEEE Transactions on Industrial Informatics, 2022.

[12] Amazon Web Services, "AWS Blockchain Whitepaper" [Online]. Available: https://d1.awsstatic.com/whitepapers/blockchain/AWS_Blockchain_Overview.pdf.