Enhancing Research Results with Programmable Blockchain Network

[1]Oluwaseyi Ajayi, and [2]Tarek Saadawi

[1]*Engineering Department, Vaughn College of Aeronautics and Engineering, NY, USA*

[2]*Department of Electrical Engineering, City University of New York, City College, NY, USA*

Oluwaseyi.ajayi@vaughn.edu saadawi@ccny.cuny.edu

***Abstract*— Since its inception in 2008, Blockchain has been proposed in different fields of study, and the research results have shown promising prospects in these areas. Despite these study results, blockchain technology has suffered some setbacks in adoption for real-life implementations. The unwillingness to adopt it stems from industries and organizations not being convinced about the proposed solutions' results. The reason is that many of the presented solution results come from simulation. While simulation results are acceptable for research purposes, industries might be skeptical about adopting a new system based only on simulation results. Researchers must present results from real-life implementations to fully convince stakeholders of the usefulness of adopting blockchain technology. However, presenting blockchain results from real-life performance is challenging because of the following significant problems: 1. Blockchain networks are customized to implement a single approach, i.e., no blockchain network can test multiple proposed implementations concurrently, and 2. There is a lack of testbeds (with enough blockchain nodes) to test proposed solutions. This ongoing work presents a Programmable Blockchain Network (PBN), which can implement multiple approaches simultaneously and a global testbed to evaluate proposed solutions in real-life scenarios. The PBN, implemented on Generic Routing Encapsulation (GRE) global testbed, uses a master-slave model for smart contracts calling to implement concurrent blockchain solutions. The preliminary result shows that the proposed solution enhances research results, convincing more industries to adopt blockchain technology.**

***Keywords*—** Authentication, Blockchain; COSM-IC, GRE, Programmable Smart Contracts, Testbed, Verification

I. INTRODUCTION

After the evolution of the Bitcoin blockchain proposed by Satoshi Nakamoto in 2008 [1], blockchain technology has widely been applied to different application domains, including Cybersecurity, Healthcare, IoT, Autonomous vehicle, finance, supply chain, education, smart grid, education, etc. [2]. As of 2023, the United States Patent Office grants over five thousand patents annually on blockchain-related solutions [3]. Blockchain technology has found its usefulness in these areas due to its data immutability capability, distributed ledger technology, and tamperproof ability, guaranteeing data integrity, confidentiality, and availability. Blockchain also eliminates trusted third-party involved in transaction processes. This trusted party can serve as a single point of failure and expose the transaction process to cyberattacks such as man-in-the-middle, leading to data manipulation, injection, and deletion.

Over the years, different researchers have proposed diverse solutions to solve various problems [4-7]. For instance, the authors in [4] described the implementation of the Blockchain Consortium to prioritize diabetes patients' healthcare in pandemic situations. [5] described the application in cybersecurity, while in [6], the authors proposed a new energy internet that described the usage of Blockchain in the smart grid. The authors in [7] described its application in transportation. Based on these results, blockchain technology has shown to be a great prospect. However, despite these research results' great potential, blockchain technology must improve its adoption for real-life implementations.

The unwillingness to fully adopt blockchain technology by industries and organizations can be attributed partly to the fact that research results should have fully covered the research solutions' behavior in real life. This notion stems from the fact that many proposed solutions are obtained from software simulation. While our work does not condemn software simulations and affirms that simulation results are acceptable as proposed solutions, industries and organizations might be skeptical about adopting and implementing a new solution based on software simulation results only. Researchers should present the real-life implementation and software simulation results to further convince stakeholders, especially in adopting blockchain technology for real-life applications. Delivering real-life implementation results will strengthen the research results, improving the overall practicality of blockchain technology.

Implementing a blockchain solution in real-life scenarios is challenging because of the following significant problems.

i. There is a need for more testbeds (with many blockchain nodes) to test the proposed solution. One of the challenges facing implementing a blockchain solution is the need for more readily available testbeds to evaluate proposed approaches. Although most Cloud Services Providers (CSP), such as AWS, Azure, and IBM, now offer blockchain nodes as part of the services rendered, CSP has two problems. a. Cost: Subscribing to these cloud services is enormous, especially when you must keep it running for a long time. b. The network provided by the cloud service providers did not mimic a real-life scenario. The nodes in their data centers are connected via high-speed ethernet connections. Implementing a solution on such a high-speed connection might make things look good, but deploying the solution to a real-life situation might turn otherwise.

ii. A blockchain network is customized to implement a single approach, i.e., no blockchain network can test multiple proposed implementations concurrently.

To alleviate these problems and improve the adoption of blockchain technology in real-life situations, This work presents a Programmable Blockchain Network (PBN). The contributions are classified as follows:

- To develop a Programmable Blockchain Network that can simultaneously support the evaluation of multiple approaches.
- The developed solution allows an API call to an existing blockchain network.
- The architecture employs a novel master-slave smart contract model for diverse solution implementations.
- The proposed solution introduces a novel Hierarchy Byzantine Fault Tolerance (HBFT) protocol for delegation selection to assign nodes to different assignments.
- The proposed solution allows the real-time addition of smart contracts, transaction verification, and authentication.
- The solution also uses a new protocol to select the miners for implementation randomly.
- The solution presents a global testbed to evaluate proposed solutions in real-life scenarios.

The remainder of this paper is organized as follows: background and related works on blockchain implementations are discussed in Section II. Section III explains the proposed architecture. Section IV presents the preliminary results. Finally, Section V presents the conclusions of this paper and possible future works.

II. BACKGROUND AND RELATED WORKS

Blockchain technology was first explained in [1] as a technology behind Bitcoin. Some of the characteristics of this Bitcoin blockchain were unfit for other businesses. Hence, private blockchain platforms were developed to be useful for businesses. In this type of network, the number of nodes can be controlled (i.e., only permissioned nodes have the privilege to participate). Also, transactions are processed at a much faster rate. The blockchain network can be considered an append-only, public ledger that keeps track of participant transactions. It is a decentralized, transparent, and chronological database of transactions. The Blockchain stores the transactions that have occurred in the network [8,9]. Each transaction in the public ledger is verified by consensus (an agreement among all participating nodes) of the participants in the system. Once the transaction is confirmed, it is impossible to mutate/erase the records. The Blockchain contains a certain and verifiable record of every transaction ever made [1]. The data in the Blockchain (i.e., transactions) is divided into blocks. Each block is dependent on the previous one (parent block). Every block stores some metadata and the hash value of the last block. Therefore, every block has a pointer to its parent block. This is how blocks are linked, creating a chain of blocks called a Blockchain. The system in which a blockchain serves as the database comprises nodes or workers. These workers are responsible for appending new blocks to the Blockchain. A new block can only be appended after nodes reach a consensus. The protocol regulates how the validity of transactions is determined and how the nodes compute new blocks. Blockchain eradicates the needed trust for third parties because all nodes reach a consensus on every transaction [10,11]

Blockchain technology has been proposed in different areas in recent years. For instance, the authors in [13] proposed a blockchain model that facilitates machine-to-machine (M2M) interaction and frames an electricity market in the context of demand requests in smart grids. Their work used Blockchain to record data derived from the power flow calculation model and electricity price customization. They used smart contracts to store transaction data and transfer assets automatically. They established a power flow calculation for a 34-node master-slave control island microgrid operation system. The power flow was calculated, and an optimal generator work adjustment was used. They presented a scenario where the power management system and generator could actively adjust the power generation trading with each other over a blockchain. The simulation result showed that the scenario verified the feasibility of the method. An Enhanced Proof-of-Work (E-PoW) consensus protocol was proposed in [14] to improve data security and privacy in healthcare, thereby reducing the bandwidth and improving efficiency. E-PoW is a lightweight consensus protocol for IoT devices in healthcare systems. The simulated result showed that the proposed E-PoW performed better in efficiency and bandwidth than the existing PoW.

In [15], the authors proposed a novel blockchain-empowered platoon communication scheme for vehicular safety applications. In their work, they formed decentralized vehicles into a platoon assisted by a pre-established blockchain-based security system. The platoon communication was then utilized to update the resource scheduling scheme enabled by the Blockchain on the attending vehicles. Monte Carlo simulation was employed to compare the results between the proposed scheme and the semi-persistent scheduling (SPS) scheme specified by the current standard. The simulation results showed that the proposed method outperformed the SPS scheme by decreasing the collision probability by at least 40% while reducing the average scheduling delay by at least 30%. A blockchain-based access control system was proposed in [16]. This work used the CP-ABE algorithm to control data encryption and access rights. The proposed model was characterized by integrity, openness, and verifiability, which could help data distribution to ensure data security and privacy issues. The result analyzed the feasibility of the scheme and gave the formula proof.

The proposed solutions' results have proven efficient and effective in all highlighted areas. However, despite the promising results from diverse applications, Blockchain adoption in real-life situations has remained the same. Much of the efforts are being shifted to proposing new solutions without concrete, real-life implementation. This work presents an implementation platform that evaluates real-life applications to enhance the research results. Hence, the motivation for this work.

III. THE PROPOSED ARCHITECTURE

The proposed architecture, built on the Ethereum blockchain platform, features a master-slave smart contracts model. The architecture was first implemented on KYUSHU-CCNY Generic Routing Encapsulation (GRE) Tunnel (more on this in the next section). Ethereum was selected due to its popularity and potential wide variety of applications. Ethereum is an open-source blockchain platform featuring smart contracts. Smart contracts are programs stored on the chain and run by all participants [17]. They can be executed upon predefined conditions [18]. The architecture is divided into three different stages, as shown in Fig. 1

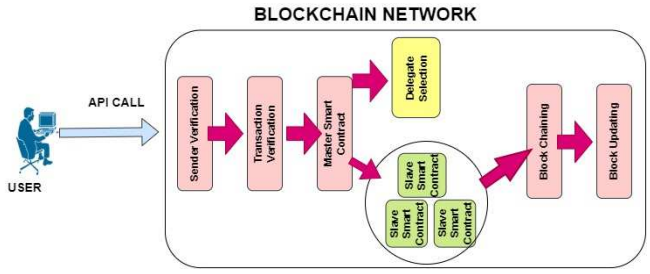


Fig. 1 The Proposed Architecture

A. API Calls and Transaction Preparation

Application Programming Interface (API) calls interact with blockchain networks from users in any location. An API call is defined as a request to use an existing smart contract or a request to add a new smart contract. When a user makes an API call, the request is built into a transaction and submitted to the blockchain network. The blockchain network verifies the call and the sender for every API call. A tag accompanies each transaction from a user. This tag contained verification information about the sender and the call format.

B. Transaction Analysis

The master smart contract verifies the transaction (API call) submitted by a user. In our architecture, the smart contract handles four functions:

1. *Sender verification:* The purpose of verifying the API sender is to ensure that only registered senders can perform experiments on the blockchain network. We implement a policy that forces users to register on the network before making an API call. When receiving an API call, the master smart contract retrieves the accompanying tag and invokes the code snippet that compares it with the stored information. The information verified includes the transaction account and digital signature. If the sender verification is successful, the algorithm invokes the transaction verification code [19].
2. *Transaction verification:* Transaction verification is necessary because there is a need to identify and thwart any malicious transaction or activities in the transaction. Furthermore, ensuring the integrity and consistency of the submitted transaction before it is invoked is pertinent. In this architecture, the master smart contract behaves like a firewall that analyzes every ingress transaction to determine the next step. An API call performs two functions: 1. Calling for an existing smart contract in an ongoing implementation and 2. Introducing a new smart contract for a new implementation [20]. In each case, the master smart contract analyzes the transactions and performs the following two functions.

i. *Call a slave contract*: This occurs when a user resumes an ongoing implementation or wants to use an existing smart contract. Based on the verification information retrieved from the transaction, the master smart contract calls the appropriate slave contract so the user can perform the implementation.

ii. *Add a slave contract:* This occurs when users attempt to commence new projects. In this case, the existing contracts might not fit perfectly to their implementation. When an API call is made to the blockchain network, the master smart contract analyzes it, adds the new smart contract to the list of slave contracts, and provides its information to the user. The contract information must be in the API call for a user to use any contract. Based on this information, the master smart contract calls the appropriate smart contract. The flowchart in Fig. 2 describes the function of the master smart contract.

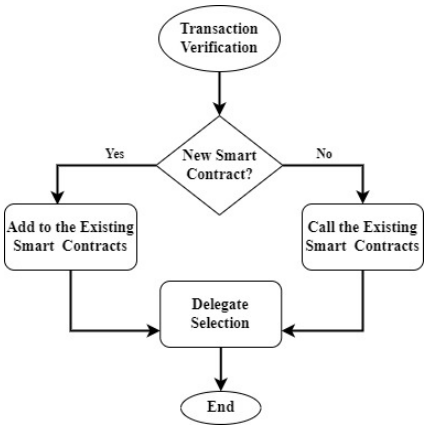


Fig. 2. Flowchart of Master smart contract function

3. *Delegate Selection:* After a successful verification, the master smart contract performs the delegation selection process. Delegate selection is when the master smart contract randomly picks the miners for an implementation. This process allows the nodes to divide, work independently on different implementations, and keep a unified ledger. This process enables the architecture to carry out the implementation of various applications concurrently. The smart contract chooses delegates using the Hierarchy Byzantine Fault Tolerance (HBFT) protocol. The HBFT is based on node reputation [21]. Unlike [21], a multilayer hierarchy structure is designed to improve scalability by assigning nodes to different assignments. Each node only needs to exchange messages within its group, reducing communication complexity between

nodes. Specifically, a reputation model is proposed to distinguish normal nodes from malicious ones by a punishment and reward mechanism. A random selection mechanism is applied in the selection of the lead miners. The mechanism ensures the blockchain network's security with unpredictability and randomicity characteristics.

C. Transaction Validation

The pending transaction from the verification stage is built into a block by the nodes after the verification stage is successful. The block is broadcast among the delegates for validation. The delegates within the group receive the block and work to validate it through a process that requires consensus from all authorized nodes. This was achieved using Proof of Stake (PoS). The PoS algorithm works on the principle that for a node to participate in the validating process, it must "stake" some of its cryptocurrency as collateral. Miners are selected randomly to confirm the transactions and validate the block information. The security of this algorithm comes if a miner validates a transaction that is considered invalid by other miners, such miner loses a portion or all its stake, and its reputation goes down.

D. Transaction Updating

The new block is chained to the Blockchain when the transactions have been validated. The transaction address is issued to the transaction owner (sender). The blockchain's current state (i.e., the update of the new block) is broadcast to every node in the network. Every node in the blockchain network receives a copy of the recent update. Hence, the architecture keeps one public ledger.

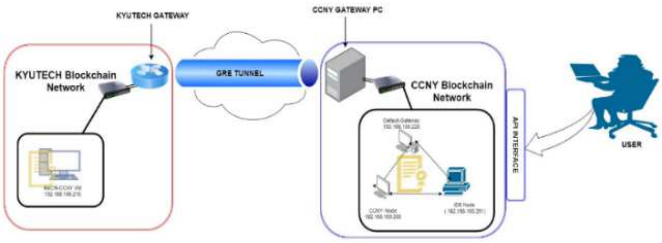
IV. IMPLEMENTATION AND RESULTS

The proposed architecture is being implemented on the transpacific CCNY-KYUSHU Generic Routing Encapsulation (GRE) Tunnel, as shown in Fig. 3. The KYUTECH-CCNY GRE tunnel is a National Science Foundation (NSF) funded project to define the Internet's future. The GRE tunnel is a virtual private network connection between the locations of Kyushu Institute Japan (KYUTECH) and CUNY City College (City College of New York, CCNY), USA. It originates from CINT lab, CCNY, built through the internet 2 to the JGN network in Seattle and from the JGN network to terminate at Kyushu Institute in Japan. The project addresses research challenges associated with enabling trustworthy networks, supporting the Internet of Things (IoT), which encompasses everything connected to the Internet and cyber-physical systems (CPS), a controlled mechanism monitored by computer-based algorithms, and provides a testbed for implementations. Fig. 5 shows a high-level connection between the City College of New York (CCNY) and Kyushu Institute of Technology (KYUTECH) through an internet tunnel [22].

A. Preliminary Result

We deployed the architecture to the transpacific GRE testbed to evaluate the performance for a near real-life situation. Here, we set up part of the blockchain network on the CCNY side while the other domain is set up at KYUTECH, as shown in Fig. 4. Each fragment contains multiple blockchain nodes that exchange information. The master smart contract runs on the blockchain and communicates the slave contracts. A user who wants to implement energy trading research in the smart grid can configure the architecture's characteristics using a custom smart contract uploaded via an API call. Suppose another user wants to implement another application, say, healthcare records verification. In that case, the user can make an API to the master smart contract described in section III-B without affecting the existing smart contracts. At any point, any of these experiments can be carried out without negatively affecting the results of others.

Fig. 4 The implementation of the proposed PBN



The preliminary result shows the architecture's performance in Fig. 5 and 6. The experiment was implemented with three blockchain nodes, as shown in Fig. 5. For a detailed

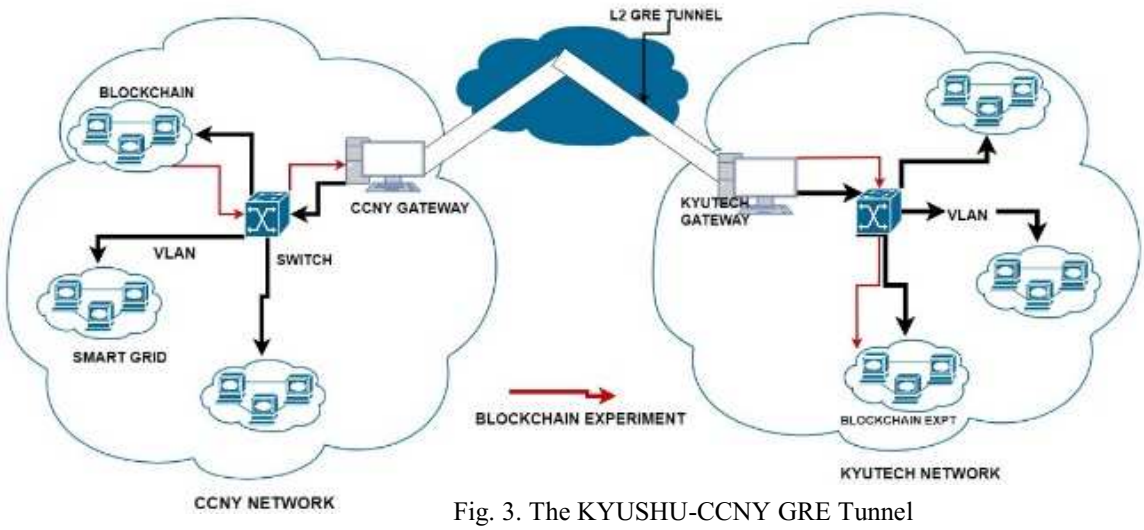


Fig. 3. The KYUSHU-CCNY GRE Tunnel

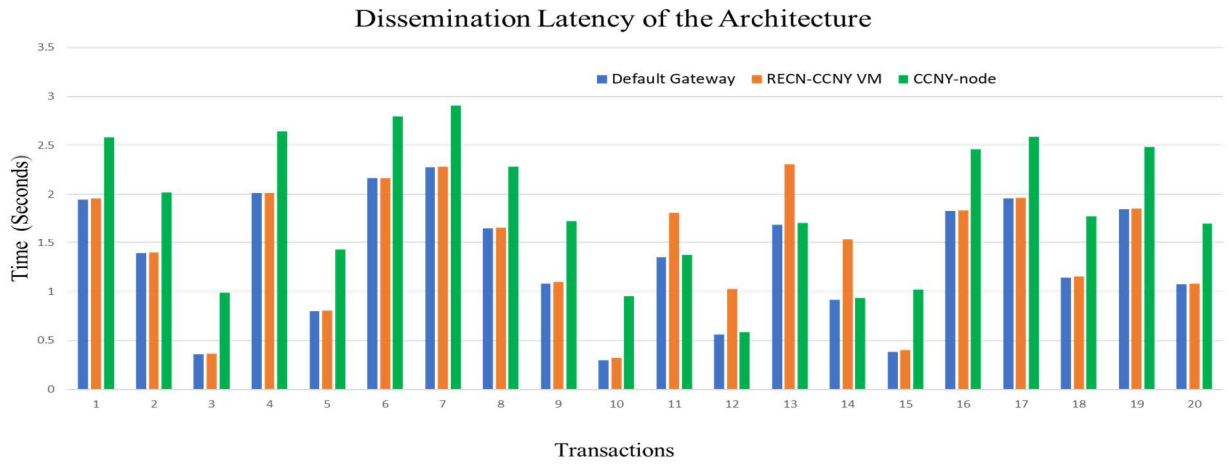


Fig. 5. The Dissemination latency for Blockchain nodes

description of the setup and implementation, visit [19] and [20]. We measured each node's dissemination latency and average dissemination latency. Fig. 5 shows the dissemination latency of blockchain nodes for twenty transactions. As explained in [23], dissemination latency is the time elapsed when a transaction is submitted to the Blockchain to when each node retrieves the information in its ledger. This time comprises verification time, commit and mining time, new block broadcasting time, and time taken to recover it from their ledger. Fig. 6 shows the average dissemination time of each node. We observe that the architecture's average response time is low (less than 2 seconds), even for a node located in Japan (RECN-CCNY VM). For further work, we will repeat the experiment with more blockchain nodes on each side of the testbed and compare the result to what was obtained initially.

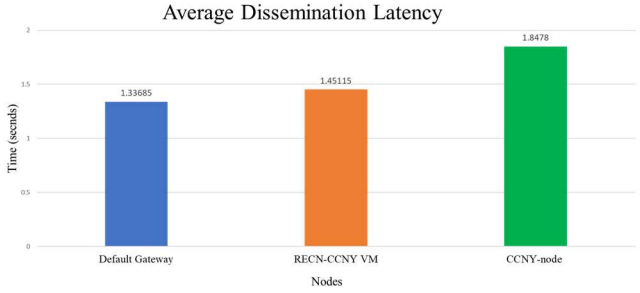


Fig. 6. The average dissemination latency of blockchain nodes

V. CONCLUSION

The proposed architecture focuses on enhancing the research result to improve the wide-range industrial adoption of blockchain technology. In this paper, we proposed a Programmable Blockchain Network (PBN) that helps blockchain researchers evaluate their proposed real-life solutions. The solutions presented a novel master-slave smart contracts concept for transaction verification, existing smart contract calling, and new smart contract addition. The model introduced a Hierarchy Byzantine Fault Tolerance (HBFT) protocol for delegation selection. Also, the method presented a novel platform for evaluating multiple unrelated solutions concurrently. The methodology describes the real-life implementation of the solution on a Kyushu-CCNY GRE testbed. The preliminary result shows that the proposed solutions can strengthen research results in blockchain applications, hence, improving the wide-range adoption of the technology.

The next phase of this ongoing research is to implement the architecture on a COSMOS Interconnecting Continents (COSM-IC) global testbed (Fig. 7). The COSM-IC is a National Science Foundation (NSF) sponsored global testbed spanning Asia, Europe, South and North America continents. The testbed proposes to develop capabilities that will enhance unique multi-technology and software-defined wireless, optical, and edge cloud network to perform a groundbreaking international collaborative experiment. The goal will be achieved by leveraging the COSMOS [RSZ+20, COS20] and ORBIT [RSO05, BCL14] testbeds' interfaces with the PEERING [SAC+19] and FABRIC [FAB20] testbeds and adding connections to leading testbeds worldwide, including CPQD (Brazil) [CPQD20], Kyutech U./StarBED (Japan) [Sta20], OneLab/NITOS (EU/Greece) [One20, NIT20], and CONNECT and Pervasive Nation at Trinity College Dublin (EU/Ireland). Upon completing this global testbed, the architecture will be deployed and evaluated.

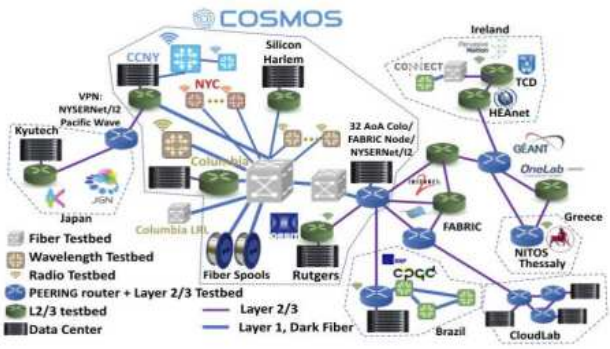


Fig 7. COSM-IC Testbed [22]

ACKNOWLEDGMENT

This work is supported by NSF Grant, INRC Testbed: COSMOS Interconnecting Continents.

REFERENCES

[1] S. Nakamoto (2008) Bitcoin: a peer-to-peer electronic cash system, http://bitcoin.org/bitcoin.pdf

[2] M. A. Khan and K. Salah, "IoT security: Review, blockchain solutions, and open challenges," Future Generation Computer Systems, vol. 82, pp. 395–411, 2018.

[3] https://www.bitlaw.com/blockchain/blockchain-patents.html 4/26/2023

[4] G. Subramanian and A. Sreekantan Thampy, "Implementation of Blockchain Consortium to Prioritize Diabetes Patients' Healthcare in Pandemic Situations," in IEEE Access, vol. 9, pp. 162459-162475, 2021, doi: 10.1109/ACCESS.2021.3132302.

[5] K. Cremona, D. Tabone and C. De Raffaele, "Cybersecurity and the Blockchain: Preventing the Insertion of Child Pornography Images," 2019 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), Guilin, China, 2019, pp. 197-204, doi: 10.1109/CyberC.2019.00042.

[6] I. Dimobi, M. Pipattanasomporn and S. Rahman, "A Transactive Grid with Microgrids Using Blockchain for the Energy Internet," 2020 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT), Washington, DC, USA, 2020, pp. 1-5, doi: 10.1109/ISGT45199.2020.9087739.

[7] Z. Farr, M. Azab and E. Samir, "Blockchain-based cooperative autonomous detection of suspicious vehicles," 2020 11th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), Vancouver, BC, Canada, 2020, pp. 0188-0192, doi: 10.1109/IEMCON51383.2020.9284881.

[8] M Signorini and M Pontecorvi, W Kanoun, and R Di Pietro, "BAD: a Blockchain Anomaly Detection solution" arXiv:1807.03833v2, [cs.CR] 12 Jul 2018

[9] Ingo Weber, Vincent Gramoli, Mark Staples, Alex Ponomarev, Ralph Holz, An Binh Tran, and Paul Rimba. 2017. On Availability for Blockchain-Based Systems. In SRDS'17: IEEE International Symposium on Reliable Distributed Systems.

[10] T. Ahram, A. Sargolzaei, S. Sargolzaei, J. Daniels, and B. Amaba. "Blockchain Technology Innovation". 2017 IEEE Technology & Engineering Management Conference (TEMSCON), 2017

[11] T. Golomb, Y. Mirsky, and Y. Elovici " CIoTA: Collaborative IoT Anomaly Detection via Blockchain" arXiv:1803.03807v2, [cs.CY] 09 Apr 2018

[12] S. Pongnumkul, C. Siripanpornchana and S. Thajchayapong, "Performance Analysis of Private Blockchain Platforms in Varying Workloads," 2017 26th International Conference on Computer Communication and Networks (ICCCN), Vancouver, BC, Canada, 2017, pp. 1-6, doi: 10.1109/ICCCN.2017.8038517.

[13] X. Wu, B. Duan, Y. Yan and Y. Zhong, "M2M Blockchain: The Case of Demand Side Management of Smart Grid," 2017 IEEE 23rd International Conference on Parallel and Distributed Systems (ICPADS), Shenzhen, China, 2017, pp. 810-813, doi: 10.1109/ICPADS.2017.00113.

[14] G. S. Gunanidhi and R. Krishnaveni, "Improved Security Blockchain for IoT based Healthcare monitoring system," 2022 Second International Conference on Artificial Intelligence and Smart Energy (ICAIS), Coimbatore, India, 2022, pp. 1244-1247, doi: 10.1109/ICAIS53314.2022.9742777.

[15] L. Cao and H. Yin, "A Blockchain-Empowered Platoon Communication Scheme for Vehicular Safety Applications," 2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall), Norman, OK, USA, 2021, pp. 1-6, doi: 10.1109/VTC2021-Fall52928.2021.9625342.

[16] H. Bowen, L. Yi, F. Li, D. Xinhua and C. Ping, "Blockchain-based Access Control Data Distribution System," 2019 IEEE 5th International Conference on Computer and Communications (ICCC), Chengdu, China, 2019, pp. 1231-1236, doi: 10.1109/ICCC47050.2019.9064149.

[17] I. Weber et al., "On Availability for Blockchain-Based Systems," 2017 IEEE 36th Symposium on Reliable Distributed Systems (SRDS), Hong Kong, China, 2017, pp. 64-73, doi: 10.1109/SRDS.2017.15.

[18] P. Zhang, M. A. Walker, J. White, D. C. Schmidt and G. Lenz, "Metrics for assessing blockchain-based healthcare decentralized apps," 2017 IEEE 19th International Conference on e-Health Networking, Applications and Services (Healthcom), Dalian, China, 2017, pp. 1-4, doi: 10.1109/HealthCom.2017.8210842.

[19] O. Ajayi and T. Saadawi, "Detecting Insider Attacks in Blockchain Networks," 2021 International Symposium on Networks, Computers and Communications (ISNCC), Dubai, United Arab Emirates, 2021, pp. 1-7, doi: 10.1109/ISNCC52172.2021.9615799.

[20] O. Ajayi and T. Saadawi, "Blockchain-Based Architecture for Secured Cyber-Attack Features Exchange," 2020 7th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2020 6th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom), New York, NY, USA, 2020, pp. 100-107, doi: 10.1109/CSCloud-EdgeCom49738.2020.00025.

[21] X. Wang and Y. Guan, "A Hierarchy Byzantine Fault Tolerance Consensus Protocol Based on Node Reputation," *Sensors*, vol. 22, no. 15, p. 5887, Aug. 2022, doi: 10.3390/s22155887

[22] O. Ajayi, H. Huseynov, T. Saadawi, M. Tsuru and K. Kourai, "Transpacific Testbed for Real-Time Experimentation," 2021 IEEE 4th 5G World Forum (5GWF), Montreal, QC, Canada, 2021, pp. 505-510, doi: 10.1109/5GWF52925.2021.00095.

[23] O. Ajayi, M. Cherian and T. Saadawi, "Secured Cyber-Attack Signatures Distribution using Blockchain Technology." 2019 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), New York, NY, USA, 2019, pp. 482-488.