Bidirectional Authentication for Safe Ridesharing Empowered by Permissioned Blockchain

Wanxin Li[1] Collin Meese[2] Mark Nejad[2] Hao Guo[3*] Jie Zhang[1*]
[1]School of Advanced Technology, Xi'an Jiaotong-Liverpool University, China
[2]Department of Civil and Environmental Engineering, University of Delaware, USA
[3]School of Software, Northwestern Polytechnical University, China
{wanxin.li, jie.zhang01}@xjtlu.edu.cn, {cmeese, nejad}@udel.edu, haoguo@nwpu.edu.cn

2025 IEEE Global Blockchain Conference (GBC) | 979-8-3503-6642-6/25/$31.00 ©2025 IEEE | DOI: 10.1109/GBC60041.2025.11134467

***Abstract*—Ridesharing and on-demand mobility systems offer societal benefits that include reduced traffic, lower parking demand and less environmental impact from vehicle usage. However, the problem of user impersonation has compromised the safety of both riders and drivers, sometimes ending in fatal tragedy. To address the safety concerns resulting from user impersonation, this paper proposes a blockchain-based and zero-knowledge approach for decentralized and privacy-preserving identity verification in ridesharing. The proposed permissioned blockchain facilitates our privacy-aware verification scheme and provides fine-grained access control policies to protect on-chain trip records. We developed the proposed system on the Hyperledger Fabric platform, with Chaincode smart contracts and Hyperledger Ursa cryptographic library. To measure the performance of the system, we conduct extensive experiments utilizing the Hyperledger Caliper benchmark tool. Our results show that the zero-knowledge proof module can perform the privacy-preserving identity verification at the millisecond level while the blockchain network offers low latency and high throughput for transactions. The non-resource-intensive authentication scheme and the proposed secure-by-design blockchain with access control policies make the proposed approach fitting for application in real-world ridesharing environments.**

***Index Terms*—Blockchain, data privacy, identity verification, ridesharing.**

I. Introduction

Ridesharing services have gained significant awareness during the last decade as a practical approach for enhancing societal mobility. Additionally, a recent parking study shows that vehicles are not in use for an average of 95% of their lifetimes [1], providing many opportunities for increasing current vehicle utilization through ridesharing applications. In conjunction with emerging connected and autonomous vehicle technologies, ridesharing systems can provide additional societal benefits, including reduced energy use, fewer emissions [2], lower parking demand and decreased traffic congestion [3].

Nevertheless, protecting the safety of users (both riders and drivers) is an ongoing challenge in ridesharing systems, making it difficult to grow adoption and realize the associated benefits. In the existing ridesharing systems, both riders and drivers typically identify each other by verbal communication, which presents safety concerns in the event that a malicious actor seeks to infiltrate the system disguised as a trusted user. Furthermore, it is a challenge to provide the identity verification function while also respecting users' privacy.

As stated in a recent report by Uber, there were over 5,900 cases of assault-related incidents and even nine assault-related deaths on their ridesharing platform in the US from 2017 and 2018 in total [1]. Consequently, the user impersonation problem presents a serious threat to users' safety in existing ridesharing systems due to the lack of a secure and privacy-preserving identity authentication protocol.

Recently, novel blockchain architectures have been proposed for decentralized two-sided sharing economies, including ridesharing systems [4] [5]. Initially proposed in 2008 as the core technology for Bitcoin [6], blockchain represents a distributed and decentralized network technology allowing anonymous networked peers to reach consensus on the state of an immutable digital ledger. For a recent survey on consensus protocols for the blockchain networks, we refer the reader to [7]. While the blockchain ledger records the current and historical states of digital assets, the smart contracts define the executable logic of digital assets on the ledger. Two critical properties of blockchain technology, data provenance and data immutability, provide potentials for designing secure and decentralized identity verification protocols for ridesharing. Nevertheless, the privacy-preserving and low latency requirements of the bidirectional authentication in ridesharing environments cannot be addressed by the existing blockchain-based approaches.

Public blockchain systems show significant privacy concerns when sensitive user information is involved. Any networked participant can read the entire ledger in a permissionless blockchain system (e.g., Bitcoin and Ethereum) due to the transparency-by-design property. On the other side, a permissioned blockchain system (e.g., Hyperledger Fabric) allows for programmable access controls, making it suitable for constructing a blockchain-based ridesharing system with the consideration of data ownership and data privacy. However, the access control policy serves to later protect the sensitive information stored on the ledger, but it does not provide a privacy-preserving approach for verifying a user's identity at the beginning.

To address user impersonation in ridesharing, we propose a secure, decentralized, and privacy-preserving identity verification system. Our approach integrates a zero-knowledge verification protocol with a permissioned blockchain network, verifying rider and driver identities without disclosing sensitive information. The blockchain also stores trip records and enforces programmable access control policies for data access and provenance. Developed on Hyperledger Fabric and Ursa, our system ensures privacy-preserving verification and was benchmarked using Hyperledger Caliper to evaluate performance in proof generation, verification time, transaction latency, and resource consumption.

[1]https://www.uber.com/us/en/about/reports/us-safety-report/

II. Related Work

Many studies have proposed novel designs of blockchain technology in a wide array of subject areas outside of finance, including healthcare [8], internet of things [9] [10], smart cities [11] [12], and artificial intelligence [13] [14]. Recently, blockchain technology has also been investigated in intelligent transportation systems, given the distributed and decentralized nature of the internet of vehicles (IoV) [15]–[18]. Li et al. [19] propose a blockchain-based method to protect emerging intelligent traffic signal systems against malicious attackers, and extend their study into an environment of multiple vehicular networks [20]. In [21], Lin et al. showcase a blockchain-based deep reinforcement learning architecture for reliable spatial crowdsourcing in the software-defined internet of vehicles, which improves performance and guarantees data privacy. In addition, recent studies have proposed designs for end-to-end decentralized ridesharing solutions which leverage existing public blockchain systems. Baza et al. [4] introduce a ridesharing system with proof of concept implemented on Ethereum, which is trustless and preserves users' privacy. The proposed scheme includes systems for payment, matching, and reputation management, controlled by smart contracts, which utilize zero-knowledge-range proofs to protect location information. This design differs from our proposed system because it leverages a permissionless and public blockchain network as the controller of the application, presenting additional communication overhead compared to our permissioned blockchain approach outlined in this paper. In comparison, our system also offers programmable access control policies for clients, protecting the privacy of data stored on the ledger, unlike in a permissionless blockchain network.

Comprehensive ridesharing systems atop a public blockchain were proposed in [22] and [23], which preserve privacy using pseudonymity schemes. These studies motivated our research into a design of identity verification and trip records retrieval in ridesharing, which requires no pseudonyms or exchange of private information between either party. Additionally, in [24], the authors propose a smart contract based access control and direct two-party encryption approach to privacy preservation in a decentralized ridesharing environment. That being said, the two-party encryption approach still requires the exchange of sensitive information, unlike in our scheme where the underlying identity information is never transmitted. Zhang et al. [25] propose a novel and decentralized ridesharing system, focused on the package delivery application, that preserves the location privacy of users. Their scheme introduces a hash-oriented practical Byzantine Fault Tolerance (pBFT) consensus algorithm to reduce consensus latency from minutes to about 15 seconds. However, their privacy-preserving design is rooted in the package delivery application and utilizes mailbox locations exclusively as the pickup and dropoff points, making the design application-specific.

Providing secure and privacy-preserving bidirectional identity verification in ridesharing is a challenging problem and demands new design solutions. The application requirements necessitate a privacy-aware system with high throughput and low latency, which cannot be addressed by the existing blockchain-based approaches. This paper contributes to the state of the art by proposing a fast, non-resource-intensive and privacy-preserving verification protocol managed by the proposed blockchain architecture, which also includes programmable access control policies to protect the stored ledger data.

III. System Architecture

The proposed system architecture for bidirectional and privacy-preserving authentication in ridesharing includes a verification protocol based on zero-knowledge proof and the permissioned blockchain network with access control policies. We first describe the following entities taking part in the proposed system:

- Permission Issuer: The permission issuer represents a trusted organization that issues cryptographic key pairs and identity information (e.g., driver license and state ID) to the data owners and verifiers. In reality, agencies such as the Department of Motor Vehicles (DMV) are suitable candidates to fill this role.
- Rider: A rider is a registered client user in our blockchain network. The rider possesses the issued identity information (e.g., state ID) and wants to prove his/her identity to drivers in the network without disclosing the actual information.
- Driver: A driver is a registered client user in our blockchain network. The driver possesses the issued identity information (e.g., driver licence) and needs to prove his/her identity to riders in the network without disclosing the actual information.
- Peer Node: A peer node is a network entity that hosts and maintains the blockchain ledger and performs the role of data verifier that validates the identity information from users. Peer nodes are deployed and managed by a consortium of multiple organizations including DMV and ridesharing companies, forming a decentralized but permissioned blockchain network.
- Permissioned Blockchain: The permissioned blockchain network operates as the controller of the system and provides an immutable transaction ledger for recording the zero-knowledge proof and trip records.

A. System Workflow in Ridesharing

The workflow of the proposed system architecture for safe ridesharing is depicted in Figure 1. First, both riders and drivers register themselves as clients in the blockchain system. The permission issuer (e.g., DMV) generates and distributes unique prover keys to each client and corresponding verifier keys to the blockchain peer nodes. For each client, the registration and key issuance procedure only need to be performed once. After a ridesharing service matches the two clients, both the rider and the driver use their prover keys to generate one-time zero-knowledge proofs based on their identity information, and send the proofs to any peer node in the blockchain network. The peer node then uses the verifier keys to authenticate the two proofs from the clients without revealing their identities. When the authentication is completed, the blockchain system will notify both the rider and the driver. At the same time, a smart contract is executed to record the trip details as a transaction on the ledger. Subsequently, the driver can start the trip with

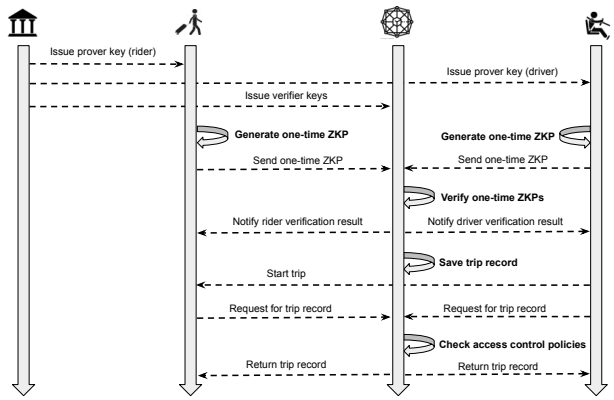

Figure 1. Workflow of the bidirectional and privacy-preserving authentication procedure in ridesharing.

the rider. Clients can also send retrieval requests for trip records through smart contracts, and the blockchain system will validate the identity information against access control policies. If the condition is satisfied, a smart contract will return the corresponding records to the client.

B. Privacy-preserving Verification Protocol

We introduce the bidirectional verification protocol for validating both riders' and drivers' identities without disclosing any sensitive information to each other. When a ridesharing service pairs a rider with a driver, both participants act as provers to prove their identities, in ZKP-based encrypted messages, to a peer node from the blockchain network that acts as a verifier. After the peer node authenticates the identities, the result is notified to both participants.

Bilinear Pairing Property. *Let G be a multiplicative cyclic group of prime order p with generator g. Let $e : G \times G \rightarrow G_T$ be a computable, bilinear and non-degenerate pairing into the group G_T. Then, we have $e(x^a, y^b) = e(x, y)^{ab}$ for all $x, y \in G$ and $a, b \in \mathbb{Z}$ because G is cyclic.*

Based on the property of bilinear pairing [26], We describe the proposed privacy-preserving verification protocol in Algorithms 1, 2 and 3. Algorithm 1 represents the key generation process, in which the permission issuer issues the prover key p_i and verifier key v_i for prover i (either a rider or a driver). Algorithm 2 describes the zero-knowledge proof generation process when the prover i first computes a hash digest h_i based on identity information m_i and generates the one-time zero-knowledge proof δ based on the hash digest h_i. Algorithm 3 presents the proof verification process that the verifier (a peer node in blockchain network) verifies the prover i without revealing the identity information m_i. In this verification process, the one-time zero-knowledge proof δ_i, hashed identity information h_i, verifier key v_i and the generator g are used for bilinear pairing check on the elliptic curve e [27].

Correctness Analysis. Assume that a prover i generates a one-time zero-knowledge proof δ_i based on the identity information m_i and sends this proof δ_i to the verifier through transaction request. We prove that the proof δ_i can be validated by Algorithm 3 as follows:

Proof.

First, the verifier key v_i is computed as:

$$v_i = g^{p_i}, \tag{1}$$

Then, a one-time zero-knowledge proof is generated as:

$$(m_i, p_i) \longrightarrow \delta_i = H(m_i)^{p_i} = {h_i}^{p_i}, \tag{2}$$

Next, verifying the proof δ_i is done by checking that iff:

$$e(\delta_i, g) = e(h_i, v_i), \tag{3}$$

According to the bilinear pairing property, we have:

$$\begin{aligned} e(\delta_i, g) &= e(h^{p_i}, g) \\ &= e(h_i, g^{p_i}) \\ &= e(h_i, v_i). \end{aligned} \tag{4}$$

■

In this way, the proof δ_i can be validated by the verifier without knowing the prover's identity information m_i. In the experiment, we choose Hyperledger Ursa library to build the generator g and elliptic curve e for bilinear pairing.

Algorithm 1: Key Generation

Input : prover i
Output: prover key p_i, verifier key v_i

1 The permission issuer selects a random $a_i \in \mathbb{Z}_p$ for prover i ;
2 The permission issuer saves the prover key as $p_i = a_i$;
3 The permission issuer computes the verifier key as $v_i = g^{p_i} \in G$;
4 The permission issuer returns p_i and v_i ;

Algorithm 2: Zero-knowledge Proof Generation

Input : identity information m_i, prover key p_i
Output: one-time zero-knowledge proof δ_i

1 The prover computes a hash digest h_i based on identity information m_i via SHA256 algorithm [28], as $h_i = H(m_i)$;
2 The prover generates the one-time zero-knowledge proof $\delta_i = {h_i}^{p_i} \in G$;
3 The prover returns δ_i ;

Algorithm 3: Zero-knowledge Proof Verification

Input : one-time zero-knowledge proof δ_i, hashed identity information h_i, verifier key v_i
Output: identity verification result r_i

1 the verifier checks **if** $e(\delta_i, g) == e(h_i, v_i)$ **then**
2 | $r_i = True$;
3 **else**
4 | $r_i = False$;
5 **end**
6 The verifier returns r_i ;

C. Permissioned Blockchain with Access Control

In this subsection, we explain the design of the permissioned blockchain network, which maintains a distributed ledger for recording transaction information, including the driver's and the rider's names, origin, destination, and price of the trip. To protect on-chain data privacy, we define access control policies and enforce the system to determine which data users are allowed to retrieve. With the access control policies deployed in the blockchain network, a user can only retrieve his/her historical transactions. Access control policies are defined with the following components:

- Participant: It represents the people or entities who are involved in the procedure of access control.
- Operation: It indicates the actions taken in the access control procedure. It can be either READ or WRITE.
- Resource: It represents the ledger data to which the access control policy applies. It can be either trip records or user profile information.
- Condition: It indicates the conditional statements over multiple variables. Combinations of multiple conditional statements are supported to serve advanced access control design.
- Action: It represents the decisive action for executing the access control procedure. It can be either ALLOW or DENY.

IV. Experiments and Evaluation

A. Experimental Setup

We developed the proposed system for safe ridesharing, which involves two primary parts interacting seamlessly: the verification module and the blockchain network. The verification module is developed utilizing the Hyperledger Ursa library. The blockchain network is built on the Hyperledger Fabric v1.2 and tested using the Hyperledger Caliper benchmark tool. We instantiate ten clients for testing, including five drivers and five riders in the blockchain network. The experiments are deployed and conducted on Ubuntu 18.04 operating system with 8GB DDR4 memory and 2.8 GHz Intel i5-8400 processor.

B. Verification Protocol

We first conducted experiments with varying secret lengths of 10, 100, and 1,000 characters and compare the results with respect to the running time of the verification protocol under each length. As shown in Figure 2, the key generation, ZKP generation and ZKP verification times are constant, even when varying the secret length. This is because our ZKP-based verification protocol hashes the secret message m to a fixed-length value of size 256-bits, by leveraging the SHA256 algorithm [28], before the proof is generated.

Consequently, the results show that key generation, ZKP generation and ZKP verification times are independent of the length of the secret message. This property provides our scheme the flexibility for verifying a myriad of secret messages (e.g., social security number and government identification number) without hindering performance or security. Verifying a proof requires more time when compared to proof generation because verifying the proof necessitates computing two pairings on the elliptic curve (Algorithm 3).

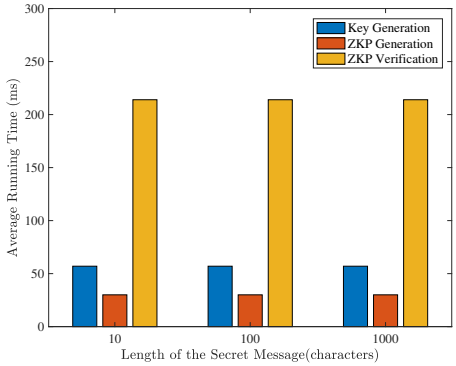

Figure 2. Comparison of key generation, ZKP generation and ZKP verification average running times among different secret message lengths.

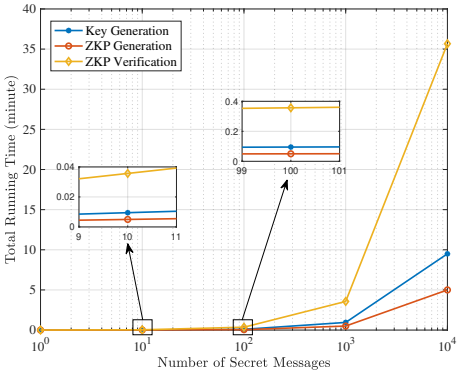

Figure 3. The relationship between total running time and the number of secret messages from key generation, ZKP generation and ZKP verification phases.

We then analyze the performance of the ZKP-based verification protocol by increasing the number of secret messages into a larger scale. Figure 3 shows the total key generation, ZKP generation and ZKP verification times by changing the number of secret messages from 1 to 10, 100, 1,000 and 10,000. As a result, the proposed verification protocol is efficient and able to handle 10,000 secret messages in 5 minutes at the prover (rider or driver) level and 36 minutes at the verifier (peer node) level.

C. Transaction Throughput

The transaction throughput measures the flow rate of processed transactions, in the unit of transactions per second (tps), through the blockchain network. As indicated in Figure 4, when increasing the transaction send rate, the average transaction throughput will increase at the start and then hit peaks at 27 tps, 17 tps, and 15 tps under 1-of-any, 2-of-any, and 3-of-any endorsement policies, respectively.

The option of endorsement policy will affect the transaction throughput. For instance, as shown in Figure 5, with a fixed transaction send rate at 20 tps, the average transaction throughput will decrease by increasing the number of endorsing peers. This is because the complexity of the endorsement process is increased by more endorsing peers. In addition, we conduct tests to record the minimum, average, and maximum

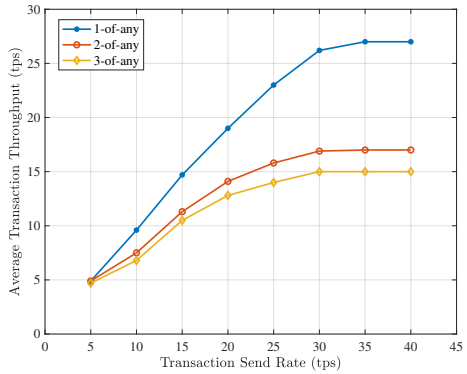

Figure 4. The relationship between average transaction throughput and transaction send rate according to different Hyperledger Fabric endorsement policies.

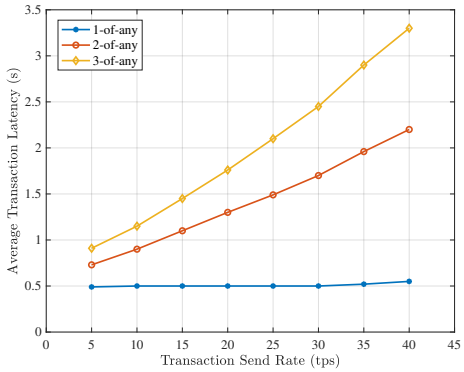

Figure 6. The relationship between average transaction latency and transaction send rate according to different Hyperledger Fabric endorsement policies.

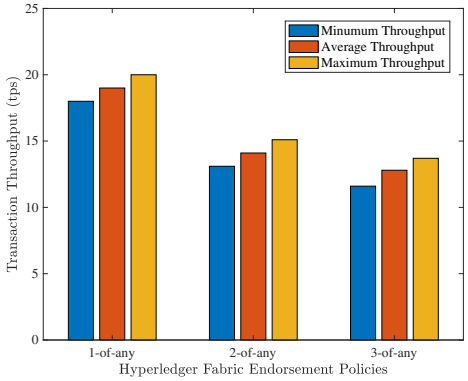

Figure 5. Comparison of minimum, average and maximum transaction throughputs among different Hyperledger Fabric endorsement policy based on a fixed transaction send rate of 20 tps.

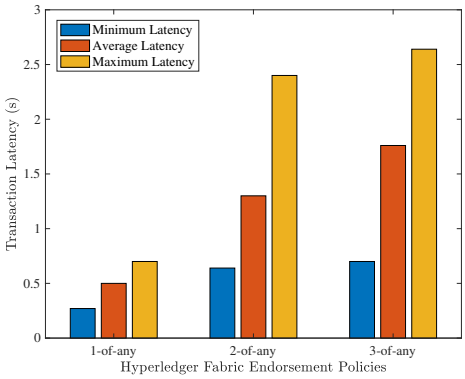

Figure 7. Comparison of minimum, average and maximum transaction latencies among different Hyperledger Fabric endorsement policy based on a fixed transaction send rate of 20 tps.

transaction throughputs. The results show that the difference between the minimum and maximum transaction throughputs is relatively small under different endorsement policies.

D. Transaction Latency

The transaction latency measures the end-to-end processing time of a blockchain transaction over the course of its lifetime, from initial submission until the final results are committed to the ledger. Illustrated in Figure 6, the average transaction latency increases significantly under the 2-of-any and 3-of-any endorsement policies as the transaction send rate is raised. That being said, the observed average transaction latency does not fluctuate under a 1-of-any endorsement policy, remaining at a constant 0.5 seconds, given the transaction send rate does not exceed 30 tps. However, if the transaction send rate exceeds 30 tps, the observed average latency will also increase under a 1-of-any endorsement policy, albeit at a significantly slower rate than the 2-of-any and 3-of-any policies. These differences can be explained by the complexity of a given endorsement policy: the 2-of-any and 3-of-any policies are significantly more complex than a 1-of-any policy and require additional overhead for communication and computation.

Besides, the chosen endorsement policy can have a significant impact on the transaction latency. For example, as shown in Figure 7, if the transaction send rate is fixed at 20 tps, an increase in the number of endorsing peers will also result in an increase in transaction latency. Moreover, we conduct multiple rounds of tests to record the minimum, average, and maximum transaction latencies. Our results show that the difference between the minimum and maximum transaction latencies will also increase when increasing the number of endorsing peers.

E. Resource Consumption

Throughout our experiments, we collected data on consensus peers' resource consumption across 1-of-any, 2-of-any, and 3-of-any endorsement policies. The results are summarized in Table I. The results show that our design uses low resources and has suitable network traffic demand for real-world applications across various IoT devices. When increasing the

Table I
RESOURCE CONSUMPTION

Endorsement	Peer	Memory	CPU	Traffic In	Traffic Out
1-of-any	peer1	203.2MB	15.17%	859.3KB	883.4KB
2-of-any	peer1	211.5MB	12.46%	979.2KB	907.2KB
	peer2	243.5MB	13.24%	967.9KB	571.8KB
3-of-any	peer1	205.7MB	11.39%	1.1MB	880.1KB
	peer2	244.5MB	11.26%	1.1MB	548.5KB
	peer3	223.4MB	12.13%	1.1MB	549.1KB

endorsement process's complexity, more messages must be sent to reach a consensus, which accounts for the increased traffic under the 3-of-any policy compared to the others.

V. Conclusion

This paper addresses the safety problem of user impersonation in existing ridesharing services by proposing a blockchain-based and zero-knowledge approach to secure, privacy-preserving and bidirectional identity verification of ridesharing clients. Our proposed design enables safe identity verification without requiring the sharing of confidential information between untrusted parties while also meeting the low latency and non-resource-intensive requirements of the ridesharing environments. We develop the proposed system and perform extensive experiments to evaluate its performance under different settings. Our results demonstrate that the proposed blockchain architecture, deployed on Hyperledger Fabric, offers high transaction throughput with low latency. Meanwhile, the proposed ZKP-based verification protocol can perform identity verification at the millisecond level, making our design suitable for use in real-world ridesharing applications.

References

[1] C. Willing, T. Brandt, and D. Neumann, "Intermodal mobility," *Business & Information Systems Engineering*, vol. 59, no. 3, pp. 173–179, 2017.

[2] L. Yan, X. Luo, R. Zhu, P. Santi, H. Wang, D. Wang, S. Zhang, and C. Ratti, "Quantifying and analyzing traffic emission reductions from ridesharing: A case study of shanghai," *Transportation Research Part D: Transport and Environment*, vol. 89, p. 102629, 2020.

[3] A. T. Moreno, A. Michalski, C. Llorca, and R. Moeckel, "Shared autonomous vehicles effect on vehicle-km traveled and average trip duration," *Journal of Advanced Transportation*, vol. 2018, 2018.

[4] M. Baza, N. Lasla, M. Mahmoud, G. Srivastava, and M. Abdallah, "B-ride: Ride sharing with privacy-preservation, trust and fair payment atop public blockchain," *IEEE Transactions on Network Science and Engineering*, pp. 1–1, 2019.

[5] P. Pal and S. Ruj, "Blockv: A blockchain enabled peer-peer ride sharing service," in *2019 IEEE International Conference on Blockchain (Blockchain)*, 2019, pp. 463–468.

[6] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system (white paper)." [Online]. Available: https://bitcoin.org/bitcoin.pdf

[7] Y. Xiao, N. Zhang, W. Lou, and Y. T. Hou, "A survey of distributed consensus protocols for blockchain networks," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 2, pp. 1432–1465, 2020.

[8] P. P. Ray, B. Chowhan, N. Kumar, and A. Almogren, "Biothr: Electronic health record servicing scheme in iot-blockchain ecosystem," *IEEE Internet of Things Journal*, pp. 1–1, 2021.

[9] X. Ding, J. Guo, D. Li, and W. Wu, "An incentive mechanism for building a secure blockchain-based internet of things," *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 1, pp. 477–487, 2021.

[10] F. Shamieh, X. Wang, and A. R. Hussein, "Transaction throughput provisioning technique for blockchain-based industrial iot networks," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 4, pp. 3122–3134, 2020.

[11] R. Chen, Y. Li, Y. Yu, H. Li, X. Chen, and W. Susilo, "Blockchain-based dynamic provable data possession for smart cities," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4143–4154, 2020.

[12] P. Kumar, R. Kumar, G. Srivastava, G. P. Gupta, R. Tripathi, T. R. Gadekallu, and N. Xiong, "Ppsf: A privacy-preserving and secure framework using blockchain-based machine-learning for iot-driven smart cities," *IEEE Transactions on Network Science and Engineering*, pp. 1–1, 2021.

[13] Z. Peng, J. Xu, X. Chu, S. Gao, Y. Yao, R. Gu, and Y. Tang, "Vfchain: Enabling verifiable and auditable federated learning via blockchain systems," *IEEE Transactions on Network Science and Engineering*, pp. 1–1, 2021.

[14] P. Bhattacharya, S. Tanwar, U. Bodkhe, S. Tyagi, and N. Kumar, "Bindaas: Blockchain-based deep-learning as-a-service in healthcare 4.0 applications," *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 2, pp. 1242–1255, 2021.

[15] S. Wang, D. Ye, X. Huang, R. Yu, Y. Wang, and Y. Zhang, "Consortium blockchain for secure resource sharing in vehicular edge computing: A contract-based approach," *IEEE Transactions on Network Science and Engineering*, 2020.

[16] B. Chen, D. He, N. Kumar, H. Wang, and K.-K. R. Choo, "A blockchain-based proxy re-encryption with equality test for vehicular communication systems," *IEEE Transactions on Network Science and Engineering*, 2020.

[17] W. Li, C. Meese, Z. Zhong, H. Guo, and M. Nejad, "Location-aware verification for autonomous truck platooning based on blockchain and zero-knowledge proof," in *2021 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, 2021, pp. 1–5.

[18] H. Guo, W. Li, M. Nejad, and C. C. Shen, "Proof-of-event recording system for autonomous vehicles: A blockchain-based solution," *IEEE Access*, vol. 8, pp. 182 776–182 786, 2020.

[19] W. Li, M. Nejad, and R. Zhang, "A blockchain-based architecture for traffic signal control systems," in *2019 IEEE International Congress on Internet of Things (ICIOT)*. IEEE, 2019, pp. 33–40.

[20] W. Li, H. Guo, M. Nejad, and C.-C. Shen, "Privacy-preserving traffic management: A blockchain and zero-knowledge proof inspired approach," *IEEE Access*, vol. 8, pp. 181 733–181 743, 2020.

[21] H. Lin, S. Garg, J. Hu, G. Kaddoum, M. Peng, and M. S. Hossain, "Blockchain and deep reinforcement learning empowered spatial crowdsourcing in software-defined internet of vehicles," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–10, 2020.

[22] Y. Kanza and E. Safra, "Cryptotransport: blockchain-powered ride hailing while preserving privacy, pseudonymity and trust," in *Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2018, pp. 540–543.

[23] M. Baza, M. Mahmoud, G. Srivastava, W. Alasmary, and M. Younis, "A light blockchain-powered privacy-preserving organization scheme for ride sharing services," in *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, 2020, pp. 1–6.

[24] Y. Semenko and D. Saucez, "Distributed privacy preserving platform for ridesharing services," in *Security, Privacy, and Anonymity in Computation, Communication, and Storage*, G. Wang, J. Feng, M. Z. A. Bhuiyan, and R. Lu, Eds. Springer International Publishing, 2019, pp. 1–14.

[25] X. Zhang, J. Liu, Y. Li, Q. Cui, X. Tao, and R. P. Liu, "Blockchain based secure package delivery via ridesharing," in *2019 11th International Conference on Wireless Communications and Signal Processing (WCSP)*. IEEE, 2019, pp. 1–6.

[26] D. Boneh, "The decision diffie-hellman problem," in *International Algorithmic Number Theory Symposium*. Springer, 1998, pp. 48–63.

[27] G. Frey, M. Muller, and H.-G. Ruck, "The tate pairing and the discrete logarithm applied to elliptic curve cryptosystems," *IEEE Transactions on Information Theory*, vol. 45, no. 5, pp. 1717–1719, 1999.

[28] D. Rachmawati, J. Tarigan, and A. Ginting, "A comparative study of message digest 5 (md5) and sha256 algorithm," in *Journal of Physics: Conference Series*, vol. 978, no. 1, 2018, p. 012116.