Monero Covert Communication-Enhanced Collaboration for Privacy-Preserving Queries

Beining Zhang, Hang Shen, Tianjing Wang, Guangwei Bai
College of Computer and Information Engineering, Nanjing Tech University, Nanjing 211816, China
{202161120012, hshen, wangtianjing, bai}@njtech.edu.cn

2025 IEEE Global Blockchain Conference (GBC) | 979-8-3503-6642-6/25/$31.00 ©2025 IEEE | DOI: 10.1109/GBC60041.2025.11134462

***Abstract*—Most existing user-collaborative privacy protection mechanisms in location-based services assume that collaborative group members are trustworthy and can strictly enforce collaboration rules. Such assumptions do not match reality and reduce the usability of the schemes. In this paper, a Monero covert communication-enhanced collaborative privacy-preserving framework is presented for mobile query services. Built on blockchain networks, this solution enables secure in-group interaction and behavior collection, enabling group members without trust to collaborate efficiently. A lightweight key negotiation is developed to support in-group Monero covert communication on blockchain networks to prevent in-group eavesdropping and selfish behavior. Blockchain-assisted behavior collection and evidence recording ensure efficient and secure execution of in-group collaboration. Security analysis and simulation results demonstrate the proposed scheme achieves high security and privacy levels with low communication and computational costs in queries.**

***Index Terms*—Collaborative privacy protection, Blockchain, Monero, covert communications, location-based service.**

I. Introduction

Location-based services (LBSs) provide location-related value-added services to positioning devices through mobile internet. While bringing great convenience to our lives, their extensive use also severely threatens user privacy [1]. Collaborative privacy-preserving solutions are group-centric, interfering with an attacker's observation by hiding the behavior of individual users within the group. However, a prerequisite for this collaborative model to be effective is that the group members are honest and trustworthy and follow collaboration rules. Most of the existing collaborative schemes default to such an assumption, but this assumption does not correspond to reality and reduces the usability of the scheme. In addition, an external malicious attacker may identify and analyze the collaborative relationship by listening to the communication between collaborators to trace the identity of the original querier.

An effective privacy protection mechanism must be able to deal with both external threats and internal hazards. External threats come from suspicious service providers (SP) who may infer users' privacy based on their background knowledge (e.g., query history or access pattern) [2]–[4]. Internal pitfalls come from untrustworthy group members who may be unresponsive or falsely claim that a task has been completed, or even steal sensitive information about group interactions.

Blockchain, which is derived from Bitcoin technology, is essentially a copy-state machine protocol in a Byzantine environment [5]. Participants can freely opt in or out of a Blockchain network, and they can ensure anonymity by using digital accounts. Being decentralized, auditable, and tamper-proof, blockchain technology can potentially solve the issue of the trustworthiness of collaborative users. First, the openness of data access facilitates the information interaction and data transfer of collaborative groups. Second, a blockchain network's flooding property and anonymity provide an ideal platform for information covertness within collaborative groups. Finally, the traceability of behavior on the chain allows us to construct a behavioral evidence record and a reward and punishment method to motivate members to comply with the collaboration [6], [7].

The above considerations drive the study of a blockchain network-assisted solution with challenges. Covert transmission is an effective means to prevent unauthorized group members from listening. Most existing covert transmissions on blockchain use asymmetric keys negotiated beforehand [8] to encrypt and decrypt the contents. This approach increases the risk of exposure in LBS scenarios with frequent interactions. The high computational latency of asymmetric keys is also a problem that should not be ignored.

A. Related Works

1) User-Collaborative Privacy Protection: The Collaboration can circumvent the drawbacks of centralized management's reliance on a trusted third party. In MobiCrowd [9], users search caches from neighbors before initiating a query. The query is directly published to the LBS if the required content is not found. This work provides valuable references. CTPP [10] is a virtual machine-based scheme where mobile users can communicate with multi-hop neighbors and share cache information. Chow *et al.* [11] designed a P2P (peer-to-peer) spatial anonymity strategy where users randomly select k-1 neighbors to form an anonymous set, then send the formed hidden area or any non-query user's location to the LBS. Unlike the above works assuming mutual trust among group members, P^4QS [12] focuses on collaboration between strangers. By using symmetric and asymmetric encryption, each user only obtains the required information, but at the cost of additional system overhead. For insecure decentralized systems, Jin *et al.* [13] proposed a security enhancement scheme for data sharing that maintains accountability through pseudonymous identity verification while protecting user privacy. Fewer studies have addressed blockchain-enabled privacy-preserving in

This work was supported by the National Natural Science Foundation of China under Grants 61502230 and 61501224, the Natural Science Foundation of Jiangsu Province under Grant BK20201357, and Six Talent Peaks Project in Jiangsu Province under Grant RJFW-020 (*Corresponding Author: Hang Shen*).

LBSs. Li *et al.* [14] used anonymous digital certificates and anonymous stealth zones to protect the location privacy of vehicles and records and maintain trust values through blockchain to defend against various attacks.

2) Blockchain-based Covert Communication: A covert channel based on a blockchain network is often defined as an undetectable communication channel that relies on the blockchain environment's unique carrier characteristics. Many covert channel construction methods use the blockchain transaction address as the carrier characteristic. BLOCCE [15] maps secret information to the significant bits of the blockchain transaction address. This approach provides an efficient way to construct a covert channel, but the capacity for message embedding is low. Cao et al. [16] presented a public key chain-based data embedding method that uses the transaction address generated by the public key as the message embedding carrier to improve covertness and transaction screening efficiency. Luo et al. [17] fused the transaction amount matrix and index address matrix to encode a secret message using the transaction amount. Despite the decrease in the transaction amount and transmission cost, the repeated use of the address set may weaken covertness. Monero [18] is a digital cryptocurrency that relies on blockchain, focusing more on protecting user privacy than the widely used Bitcoin. With the unconditional anonymity of Monero ring signatures, Guo et al. [19] encrypted the message and embedded it in the ring signature public key set, enhancing the hiding capacity and transmission covertness. Liu et al. [20] proposed a stored covert channel using Monero transactions as the data carrier for covert communication, detection resistance, and anonymity.

B. Contributions and Organization

In this paper, a Monero covert communication-enabled collaborative framework for privacy-preserving mobile queries is presented, which leverages Monero to build secure in-group interactions on blockchain networks, behavior collection, and traceability, enabling group members to defend against external attacks jointly. The main contributions are two folded:

- A lightweight constant round key negotiation is designed against in-group eavesdropping and selfish behavior. On this basis, an in-group Monero covert communication mechanism is developed, which enables secure and trusted in-group message interaction over blockchain networks.
- A blockchain-assisted behavioral collection and evidence recording mechanism is designed to secure in-group collaboration. The legitimacy of the behavior is verified by allowing collaborators to submit evidence of their collaboration. Meanwhile, an incentive mechanism creates a secure and trustworthy collaboration ecosystem. Security analyses and simulations show that the proposed scheme achieves high privacy while incurring low communication and computation costs.

The remainder of this paper is organized as follows. Section II describes the system model, consisting of threat models and design goals. In Section III, in-group covert communication is constructed. Section IV designs behavior collection and evidence recording. We validate the performance through simulations in Section VI. Section VII concludes this paper.

II. System Model

Consider a blockchain-based user-collaborative query scenario shown in Fig. 1. The scenario is assumed to involve three possible malicious actors.

- **Suspicious SP** may steal user privacy by leveraging background knowledge or eavesdropping.
- **Untrusted members** may deliberately not respond to legitimate query requests or not relay back query results and gain undeserved rewards.

A mobile user can store data and broadcast messages on the blockchain network. Neighboring users can spontaneously form a collaborative group to assist in queries and mutually protect privacy. The collaborative group relies on Monero covert communication on blockchain to secure in-group transmission. Evidence of behavior submitted by group members is stored on the chain to be verified and traced afterward.

A. Scheme Overview

As shown in Fig. 2, the symbols and operations in the framework of the proposed scheme are defined as follows:

- U represents the set of collaborative group members;
- o is the original query information of u;
- s is the fuzzified query information;
- u' is the designated collaborator of u, which can be selected depending on existing methods, such as reputation priority [21].

As shown in Fig. 2, the step of collaborative user querying based on Monero covert communication includes:

① When a query is required, u sends the fuzzy query information s to collaborator u' via covert communication;

② Collaborator u' sends s to the SP in his identity to shield u;

③ The SP processes the query s and provides the query result to u'. A suspicious SP can use s to infer u's identity and secret;

④ u' sends the results of the SP feedback back to u.

B. Threat Models

Consider an extreme environment where the threats are described as follows.

- **Selfishness:** Collaborators do not forward the query to the LBS or do not return the query result after receiving the query to save their resources;
- **Data Tampering:** Collaborators may publish tampered or forged queries that will tamper with the returned results, resulting in the querier not being able to get the correct query service;
- **Reward Repeat Claim Attack:** After completing a task, a malicious collaborator attempts to repeatedly claim the token rewards of the task by continuously creating new identities and accounts;
- **Inference Attack:** An attacker uses statistical or machine learning methods to infer a querier's identity and secret based on a prior disclosure.

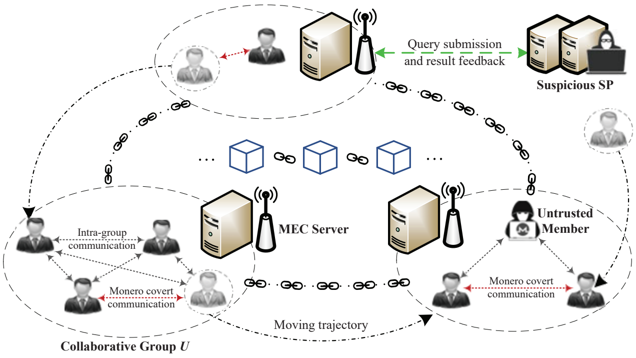


Fig. 1. Blockchain-assisted query scenario.

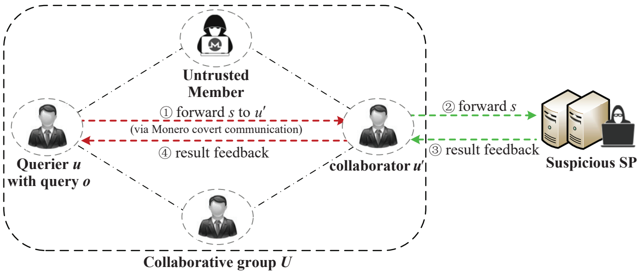


Fig. 2. Collaborative framework based on Monero covert communication.

C. Design Goals

Our scheme should achieve the following design goals.

- **Privacy Protection:** 1) It is difficult for an attacker to associate the identity of a querier with an LBS request; 2) The real identities of the members of the collaborative group are not visible to the public.
- **Security:** Communication among collaborators is hidden from external listeners, and the communication relationship is protected.
- **Trustworthiness:** Abnormalities or irregularities in in-group collaboration should be traced and punished.

III. Secure Framework for Collaboration

We construct a lightweight Monero covert communication mechanism over blockchain networks for the security and covertness of in-group interactions. The collection and evidence recording mechanism is designed for the verifiability and traceability of the collaborative behavior.

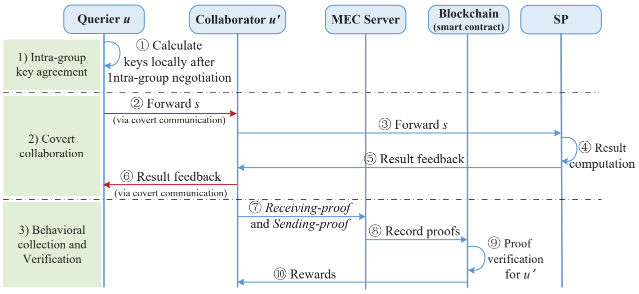


Fig. 3. In-group collaboration workflow.

A. In-group Collaboration Workflow

Fig. 3 shows the workflow of collaboration between querier u and collaborator u' on the blockchain network.

1) **In-group key agreement**: Collaborative group members negotiate the in-group key using a lightweight method. The resulting encryption keys can support in-group covert communication for a while.
2) **Covert collaboration**: To protect the confidentiality of the transmitted data, querier u and collaborator u' use Monroe-based covert communication to complete processes ② and ⑤.
3) **Behavioral collection and verification**: Once a result return is completed, u' sends receiving- and sending-proofs to the Mobile Edge Computing (MEC) server to prove that the collaboration has been completed. This evidence is recorded on the chain and can be validated afterward. Evidence of the behavior of u', who passes the verification of smart contracts, is rewarded with coins or reputation.

B. In-group Constant Round Key Negotiation

Group key agreement enables members to negotiate shared session keys in a public and untrusted network environment. To address the dynamics of collaborative groups and low-latency LBS queries, we develop a lightweight group key negotiation (LGKA) that fuses hash functions with D-H key computation [22] based on a constant round interaction framework [23]. In a collaborative group U, LGKA randomly selects a member as the controller for key agreement. For example, if member u is selected as the controller, the key agreement process proceeds as follows:

1) The elected group controller chooses a random number a_i and broadcasts $(g^{a_i}, H(g^{a_i}))$ to every member in $U \backslash \{u\}$ (g represents the generator of a mathematical group G);
2) When member i receives g^{a_i}, he verifies the integrity of g^{a_i}. When the verification passes, he randomly chooses a number b_i and sends $(g^{b_i}, H(g^{b_i}))$ to controller u;
3) When group controller u receives $(g^{b_i}, H(g^{b_i}))$ sent by member i, he verifies the integrity of g^{b_i} and generates partial key information $\prod_{j\in U\backslash\{i,u\}}^{|U\backslash\{i,u\}|} g^{a_j b_j}$, which is then sent to member i.

Members other than i and u perform the same steps as i interact with u. After all interactions are completed, the group members calculate a shared group session key as

$$k = \prod_{j\in U\backslash\{u\}}^{|U\backslash\{u\}|} g^{a_j b_j}.$$

When a user m wants to join the collaboration group, he needs to interact with u through steps 1) and 2) to make both parties calculate the key update information $g^{a_m b_m}$. After the interactions are completed, member u sends $g^{a_m b_m}$ and the original key k to the original group member and member m respectively. The group session key is updated to

$$k' = k * g^{a_m b_m}.$$

When the user leaves the collaborative group, u broadcasts an update message, $(g^{a_m b_m})^{-1}$, to other members of U other than m. The group session key is updated to

$$k' = k * \left(g^{a_m b_m}\right)^{-1}.$$

The negotiated key k can be used continuously for a period of time, supporting message encryption in subsequent covert communications.

C. In-group Covert Transmission

Monero [18] ensures that users' transactions are untraceable through ring signature technology and hides the real address of the transaction recipient by introducing a one-time address (OTA). In this subsection, we use Monero to achieve more secure in-group interactions.

Public key encryption (PKE) [24] is often used to encrypt and decrypt transmission messages in covert communication. Before each communication, the affected parties must negotiate information about PKE, which increases the risk of communication exposure. In addition, the asymmetric key used by PKE brings a long encryption and decryption delay, which is difficult to adapt to the collaboration scenario with frequent interactions.

To address the above problems, we replace the asymmetric message encryption key in the covert communication method in [20] with the symmetric key k negotiated by LGKA and construct an LGKA-based Monroe covert communication (called LGKA-MCC), which realizes the secure interaction of messages and reduces the computational burden of mobile devices. Assume that querier u intends to transmit a secret message s of length l to collaborator u'. As shown in Fig. 4, a covert communication process consists of the following.

① u encrypts s with key k (negotiated in the group beforehand) to obtain the ciphertext $C = k(s)$;

② u adds start character C_{start} and end character C_{end} to the beginning and end, respectively, of C to obtain C_{start}+C+C_{end};

③ u encodes C_{start}+C+C_{end} according to the type of data in the Monero blockchain transaction to get I;

④ u embeds I into the Monero transaction message to generate a transaction message M containing the covert information;

⑤ u sends transaction message M through the Monero blockchain network;

⑥ u' identifies transaction message M carrying the covert information from the Monero blockchain network;

⑦ u' extracts I from M;

⑧ u' decodes I according to the data type of the information carrier to obtain C_{start}+C+C_{end};

⑨ u' obtains ciphertext C from C_{start}+C+C_{end} by removing C_{start} and C_{end};

⑩ u' decrypts C with k to get secret message $s = k(C)$.

IV. Behavioral Collection and Verification

Behavior collection and verification mechanisms are designed to accommodate collaborations with frequent interactions. Evidence of behavior is recorded on the chain so that smart contracts can trace the behavior of both parties of a collaboration at the appropriate time. Users who misbehave are blacklisted. The evidence recording includes the following.

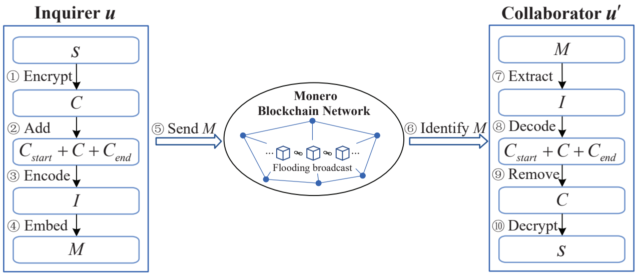


Fig. 4. Implementation framework for in-group covert transmission.

1) **Blacklisting**: The offending user's account is pulled into a blacklist stored on the chain. During the penalty period, blacklisted users are not allowed to participate in any collaboration, and they are not removed from the blacklist until the penalty expires. The penalty period is extended for users with multiple violations. Before forming a collaborative group, the smart contract verifies the identities of users who want to join, and users on the blacklist are banned from joining. After the collaboration is completed, the smart contract verifies the evidence submitted by both collaborators. Violations are traced back afterward. Once verified, an offending user is immediately blacklisted by the smart contract. Due to the ease of access to publicly stored data on the chain, retrieving the blacklist does not incur additional communication and latency.

2) **Receiving-proof**: After receiving s from u, u' submits receiving-proof containing a double signature to the smart contract. u uses his own spending key a to execute the first signature on s, generating $sign_a(s)$, and sends it to u' along with s. This relies on covert communication within the group to hide the association between u and u'. u' uses view key b to execute a second signature on $sign_a(s)$, generating $sign_b(sign_a(s))$. The smart contract verifies their signatures using the public keys $A = a \cdot G$ of u and $B = b \cdot G$ of u' to determine if u' has received s from u. The verification of the receiving-proof involves only simple signature verification and does not incur excessive computational and storage burdens.

3) **Sending-proof**: u may deny receiving the results sent by u', in which case the sending is not recognized. For this reason, after sending a query result back to u, u' submits sending-proof (containing a timestamp and block number) to smart contracts, determining whether u' has completed the sending task. If it does, a reward is given to u'; otherwise, it is blacklisted.

V. Security and Privacy Analysis

Theoretically analyzed whether the proposed solution could meet the design objectives.

- **Identity Non-correlatability:** Since the observed s has ambiguity and anonymity, an attacker needs to spend a lot of effort to guess querier u's identity.
- **Non-falsifiability:** 1) A querier has no reason to forge his obfuscated query s; 2) The attacker cannot accurately predict the querier's obfuscated output.
- **Identity Verifiability:** Smart contracts filter members based on blacklists stored on the chain, blocking malicious entities from collaborative groups.
- **Traceability**: The evidence of collaborative behaviors within a group is retained on the chain. Smart contracts can examine and trace the evidence of collaboration key results and behaviors retained on the chain.
- **Resistance to Eavesdropping Attacks:** Because covert communication special transactions are mixed in normal transactions in the blockchain network, it is difficult for an attacker to distinguish transactions carrying secrets through eavesdropping.
- **Resistance to Collusion Attacks:** Due to the ring signature in Monero, a receiver fails to backtrack the sender's identity. Collaborators cannot obtain more information about the real querier.
- **Resistance to Sybil Attacks:** Collaborators must sign messages with a real key when creating and receiving proof, making it difficult to create fake identities.
- **Resistance to Double Spending:** Smart contracts release a reward only when it detects a legitimate act that has not been paid. Once the reward is issued, the contract will prevent repeated payments for the same act.

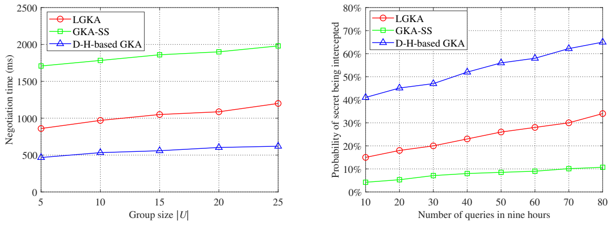
VI. Performance Evaluation

A. In-group Message Interaction Analysis

Simulation results are presented to verify the performance of the proposed in-group key negotiation, encryption, and in-group covert communication in terms of time cost and security.

1) In-group Key Negotiation Performance: This subsection examined in-group key negotiation regarding time consumption and security for different group sizes and interaction frequencies. GKA-SS [23] and D-H-based GKA [22] were chosen as benchmark methods. Signature verification in GKA-SS was omitted to allow comparison with the proposed LGKA. We constructed five collaborative groups of different sizes and collected time-consuming data generated by the collaborative groups using the three key negotiation schemes over nine hours. As shown in Fig. 5(a), the negotiation time (including the group interaction time and key computation time) for all three constant-round schemes showed a slow-increasing trend. Compared to GKA-SS, LGKA significantly reduces the negotiation time by simplifying key computation.

Fig. 5(b) reveals the impact of in-group interaction frequency on the security of group key agreement for a group size of 15. The probability of being attacked positively correlates with interaction frequency. Due to the introduction of hash functions and tighter negotiation in LGKA, the likelihood of a secret being intercepted is significantly reduced compared to D-H-based GKA. While providing security, LGKA meets requirements by simplifying key computation.



(a) (b)

Fig. 5. Impact of group size and activeness on key negotiation.

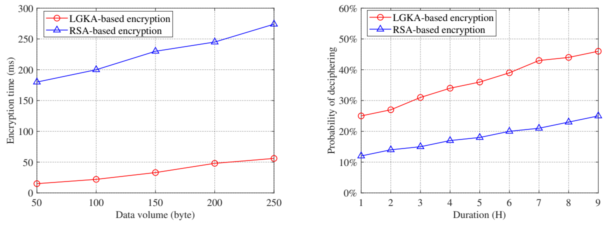
This simulation examined the effect of the transmitted data volume on the elapsed time in message encryption in a 15-person collaborative group. The RSA-based asymmetric encryption scheme [25] was chosen to compare with the LGKA-based symmetric encryption scheme. From Fig. 6(a), the encryption time of both schemes is proportional to data volume. Symmetric encryption generally involves only simple bit operations in computation, while asymmetric encryption and decryption are much more complex. Hence, the encryption time of the LGKA-based scheme is much lower than that of the RSA-based scheme.

The security of the encryption scheme was analyzed, considering a collaborative group containing an attacker. As shown in Fig. 6(b), the probability of a message being deciphered under both encryption schemes grew with time over nine hours. Asymmetric cryptographic algorithms are generally based on mathematically difficult problems and involve complex computations, making it hard for attackers to decrypt quickly. Thus, RSA-based asymmetric encryption is consistently less likely to be decrypted than LGKA-based symmetric encryption. Despite security advantages, the RSA-based scheme takes more encryption time, making adapting to scenarios with frequent collaborative interactions difficult. The designed encryption scheme can balance real-time and security, significantly reducing encryption latency at the cost of a small reduction in security.

2) In-group covert communication performance: We demonstrated the performance of the proposed LGKA-MCC in terms of the total time spent on a single communication (including channel construction duration and message transmission latency) and security, with the collaborative group size set to 15 people. PKE-based MSCCS [20] and normal communications were selected as the benchmark methods.

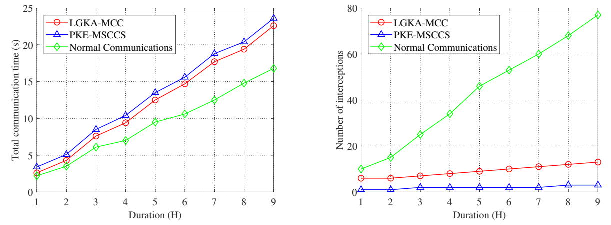
In Fig. 7(a), the time consumption caused by group members constructing Monero special transactions is slightly higher than that caused by normal communication but within acceptable limits. LGKA-MCC uses lightweight symmetric encryption, and the time it takes for message encryption and decryption is lower than that of PKE-MSCCS. LGKA-MCC provides excellent communication security at a lower time cost.

An attacker hid within the collaborative group of 15 people who could intercept the communication flow and transaction data on the blockchain network by means of traffic analysis and message eavesdropping to distinguish



(a) (b)

Fig. 6. Time cost and security of encryption scheme.



(a) (b)

Fig. 7. Time cost and security of covert communication.

special communication carrying secret information. The number of times the attacker intercepted the collaborative group under different communication methods was continuously counted, for example, within nine hours. As shown in Fig. 7(b), in-group transmission using normal communications was often intercepted. When Monero covert communication was running, special Monero transactions carrying secret information were mixed with normal transactions in the blockchain network. Since it is difficult for attackers to identify and distinguish Monero special transactions, the probability of Monero's covert communication being intercepted by attackers was much lower than that of normal communication. Compared with PKE-MSCCS, LGKA-MCC reduced the number of negotiations between the two communicating parties before transmission, reducing the risk of communication exposure.

VII. Conclusion

A Monero blockchain-enabled collaborative privacy preservation framework for mobile queries is proposed to enable collaborative group members lacking trust to collaborate effectively. A lightweight in-group key negotiation method and a Monero covert communication mechanism are designed for in-group eavesdropping and selfish behavior. Security analysis shows that the proposed in-group covert communication has high security, low communication, and computational costs. A blockchain-based verification mechanism is designed to achieve behavior verifiability and traceability. Meanwhile, a credible collaboration environment is created through incentives. Simulation results show that the solution can meet user privacy requirements at an acceptable cost. Follow-up work will explore cross-layer optimization to improve covert communication flexibility.

References

[1] H. Jiang, J. Li, P. Zhao, F. Zeng, Z. Xiao, and A. Iyengar, "Location privacy-preserving mechanisms in location-based services: A comprehensive survey," *ACM Comput. Surv.*, vol. 54, no. 1, 2022.

[2] H. Jiang, M. Wang, P. Zhao, Z. Xiao, and S. Dustdar, "A utility-aware general framework with quantifiable privacy preservation for destination prediction in LBSs," *IEEE/ACM Trans. Netw.*, vol. 29, no. 5, pp. 2228–2241, 2021.

[3] H. Shen, G. Bai, Y. Hu, and T. Wang, "P2TA: Privacy-preserving task allocation for edge computing enhanced mobile crowdsensing," *J. Syst. Archit.*, vol. 97, pp. 130–141, 2019.

[4] H. Shen, G. Bai, M. Yang, and Z. Wang, "Protecting trajectory privacy: A user-centric analysis," *J. Netw. Comput. Appl.*, vol. 82, pp. 128–139, 2017.

[5] I. Eyal, A. E. Gencer, E. G. Sirer, and R. Van Renesse, "Bitcoin-ng: A scalable blockchain protocol," in *Proc. of USENIX NSDI*, 2016, pp. 45–59.

[6] B. Zhang, H. Shen, T. Wang, and G. Bai, "Invisible man: blockchain-enabled peer-to-peer collaborative privacy games in LBSs," *Peer Peer Netw. Appl.*, pp. 1–13, 2024.

[7] H. Shen, B. Zhang, T. Wang, X. Liu, and G. Bai, "Consortium blockchain-based secure cross-operator V2V video content distribution," *Peer Peer Netw. Appl.*, pp. 1–14, 2024.

[8] F. Gao, L. Zhu, K. Gai, C. Zhang, and S. Liu, "Achieving a covert channel over an open blockchain network," *IEEE Netw.*, vol. 34, no. 2, pp. 6–13, 2020.

[9] R. Shokri, G. Theodorakopoulos, P. Papadimitratos, E. Kazemi, and J.-P. Hubaux, "Hiding in the mobile crowd: Location privacy through collaboration," *IEEE Trans. Depend. Sec. Comput.*, vol. 11, no. 3, pp. 266–279, 2013.

[10] T. Peng, Q. Liu, D. Meng, and G. Wang, "Collaborative trajectory privacy preserving scheme in location-based services," *Inf. Sci.*, vol. 387, pp. 165–179, 2017.

[11] C.-Y. Chow, M. F. Mokbel, and X. Liu, "A peer-to-peer spatial cloaking algorithm for anonymous location-based service," in *Proc. of ACM SIGSPATIAL GIS*, 2006, pp. 171–178.

[12] M. Ghaffari, N. Ghadiri, M. H. Manshaei, and M. S. Lahijani, "P^4QS: A peer-to-peer privacy preserving query service for location-based mobile applications," *IEEE Trans. Veh. Technol.*, vol. 66, no. 10, pp. 9458–9469, 2017.

[13] H. Jin and P. Papadimitratos, "Resilient privacy protection for location-based services through decentralization," *ACM Trans. Priv. Secur.*, vol. 22, no. 4, 2019.

[14] B. Li, R. Liang, D. Zhu, W. Chen, and Q. Lin, "Blockchain-based trust management model for location privacy preserving in vanet," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 6, pp. 3765–3775, 2020.

[15] J. Partala, "Provably secure covert communication on blockchain," *Cryptography*, vol. 2, no. 3, p. 18, 2018.

[16] H. Cao, H. Yin, F. Gao, Z. Zhang, B. Khoussainov, S. Xu, and L. Zhu, "Chain-based covert data embedding schemes in blockchain," *IEEE Internet Things J.*, vol. 9, no. 16, pp. 14 699–14 707, 2020.

[17] X. Luo, P. Zhang, M. Zhang, H. Li, and Q. Cheng, "A novel covert communication method based on bitcoin transaction," *IEEE Trans. Industr. Inform.*, vol. 18, no. 4, pp. 2830–2839, 2021.

[18] Y. Li, G. Yang, W. Susilo, Y. Yu, M. H. Au, and D. Liu, "Traceable Monero: Anonymous cryptocurrency with enhanced accountability," *IEEE Trans. Depend. Sec. Comput.*, vol. 18, no. 2, pp. 679–691, 2019.

[19] Z. Guo, L. Shi, M. Xu, and H. Yin, "MRCC: A practical covert channel over monero with provable security," *IEEE Access*, vol. 9, pp. 31 816–31 825, 2021.

[20] L. Liu, L. Liu, B. Li, Y. Zhong, S. Liao, and L. Zhang, "MSCCS: A Monero-based security-enhanced covert communication system," *Comput. Netw.*, vol. 205, p. 108759, 2022.

[21] T. Zhang, Y. Huo, Q. Gao, L. Ma, Y. Wu, and R. Li, "Cooperative physical layer authentication with reputation-inspired collaborator selection," *IEEE Internet Things J.*, vol. 10, no. 24, pp. 22 165–22 181, 2023.

[22] D. Augot, R. Bhaskar, V. Issarny, and D. Sacchetti, "An efficient group key agreement protocol for ad hoc networks," in *IEEE WoWMoM*, 2005, pp. 576–580.

[23] Z. Yang, Z. Wang, F. Qiu, and F. Li, "A group key agreement protocol based on ECDH and short signature," *J. Inf. Secur. Appl.*, vol. 72, p. 103388, 2023.

[24] D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in *International Conference on the Theory and Applications of Cryptographic Techniques*, 2004, pp. 506–522.

[25] K. Sharma, A. Agrawal, D. Pandey, R. Khan, and S. K. Dinkar, "Rsa based encryption approach for preserving confidentiality of big data," *J. King Saud Univ., Comp. & Info. Sci.*, vol. 34, no. 5, pp. 2088–2097, 2022.