ISSN:2790-1688 Volume-7-(2023)

Addressing blockchain privacy and efficiency challenges in mobile environments: an optimization strategy for lightweight clients and full nodes

Ke Shao[1,2,a], Wei Lv[2,b,*], Yu Li [2,c]

[1]City University Macau, Institute of Data Science, Macau 999078, China.

[2]Zhuhai College of Science and Technology,519000, China.

[a]d18092105097@cityu.mo,[b,*]luwei@zcst.edu.cn,[C]jluzhliyu@zcst.edu.cn

Abstract. In this paper, we propose an optimization strategy for privacy and efficiency challenges of blockchains in mobile environments. We propose a system model and define the corresponding threat model for the transaction query privacy problem of lightweight clients. Based on this, we design a transaction query scheme that effectively protects the privacy of lightweight clients. This scheme successfully achieves the security, privacy and completeness of transaction queries through an optimized blockchain restructuring process and privacy protection measures. Simulation experimental results show that our scheme significantly improves the privacy protection of transaction queries for lightweight clients while maintaining efficient query performance. In addition, we propose an optimization scheme for the lightweight full-node block verification efficiency problem. This scheme optimizes the verification process and introduces parallel computation, thus significantly improving the block verification efficiency. The experimental results show that our scheme achieves about 50% improvement in verification efficiency, both in terms of network size and transaction correlation. The research results in this paper not only provide a new perspective on privacy protection and efficiency improvement of blockchain technology, but also open up new applications in the mobile environment possibilities.

Keywords: blockchain,privacy,system optimization.

1. Introduction

Blockchain technology, which started as the technology underpinning Bitcoin (Bitcoin), has generated widespread research interest and has expanded beyond its initial scope of cryptocurrency applications to numerous industries and application domains, including smart contracts, supply chains, the Internet of Things, and mobile devices and services (Tapscott & Tapscott, 2016). In the mobile environment, most users will use lightweight blockchain clients rather than full-node clients that store the entire blockchain, mainly because of the limited storage and computational power of mobile devices[1]. These lightweight clients send requests to the full node when querying transactions, which can lead to the leakage of user addresses and transaction information[2]. With the popularity of mobile devices and services, how to achieve privacy protection and efficient transaction queries in lightweight clients has become an important research topic.

In addition, lightweight full nodes may encounter reduced efficiency when verifying new blocks due to the need to verify all transactions, which affects the scalability of the blockchain . To address this issue, several studies have proposed the use of sharding techniques to improve the scalability of blockchains [3], however, there are some challenges in the application of these techniques in mobile environments.

2. background and related work

2.1 Describe the applications and challenges of lightweight clients and full nodes in the mobile environment

Blockchain applications in mobile environments tend to favor lightweight clients because these devices usually have limited storage and processing power [4]. Lightweight clients store and verify

only the transaction information associated with them, rather than the entire blockchain, greatly reducing storage and computational requirements. However, this also means that lightweight clients need to rely on full nodes when executing transactions or querying history, which may lead to the disclosure of their private information. Also, the challenge for mobile devices running full nodes is how to efficiently verify and store the entire blockchain with limited resources, and how to process and service requests from lightweight clients[5].

2.2 Overview of blockchain applications and challenges in the mobile environment

Blockchain technology has a wide range of applications in the mobile environment, including mobile payments, data storage, authentication, and the Internet of Things[6]. However, these application scenarios pose new challenges to the privacy protection and efficiency of blockchain. For example, in the mobile payment scenario, how to ensure the efficiency of transactions while protecting user privacy has become an important issue. In the scenario of data storage and authentication, how to ensure the scalability of the blockchain amidst the large amount of data and requests, and how to efficiently manage and reduce the on-chain data are important issues to be studied and solved [7].

3. transaction query privacy protection for lightweight clients

3.1 Explaining the lightweight client-side transaction query privacy leakage problem

In mobile environments, lightweight clients usually do not store the full blockchain locally in order to maintain portability and high performance, but obtain transaction information by querying to the full node. When lightweight clients query transactions to the full node, transaction query privacy can be compromised. In order to address the privacy-preserving side of lightweight clients without introducing a significant communication overhead, some Bitcoin developers have proposed the use of Bloom filters[8]. Using this approach, the lightweight client embeds its bitcoin address as a keyword in the Bloom filter, which is then sent to the full node. Full nodes use this Bloom filter to filter all transactions in new blocks, and the transactions that pass this filter may contain the bitcoin addresses of lightweight clients.

However, the false alarm rate of Bloom filters may lead attackers to gradually shrink the anonymous transaction set of lightweight clients by analyzing network traffic, thus compromising users' privacy[9]. Therefore, finding a better balance between protecting user privacy and maintaining query efficiency is an important challenge for current lightweight clients.

3.2 System model, threat model

3.2.1 System model

Four components are involved in this transaction query scheme: the full node, the lightweight client, and the Bitcoin P2P network, shown in Figure 1. The lightweight client synchronizes block headers only, whereas the full node synchronizes the entire blockchain. The lightweight client is linked to the Bitcoin P2P network to synchronize blockchain data. Remote authentication establishes a secure channel between the lightweight client and the secure enclave. Lightweight clients use transaction queries to determine whether a payment was completed or to query their account balance. This is accomplished by sending the transaction's hash or bitcoin address to the secure enclave over the secure channel, and then using this information to query transactions. Secure enclaves reorganize block-chain data before processing lightweight clients' requests to enable efficient and privacy-sensitive transaction searches.

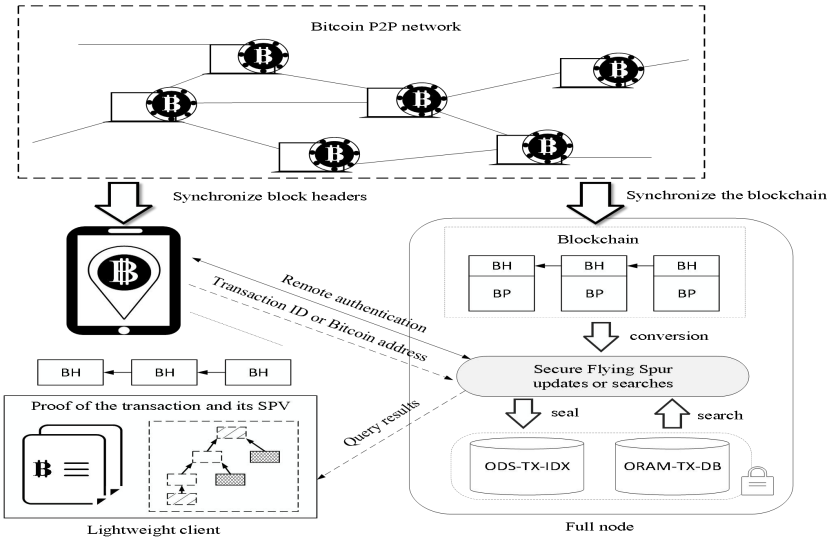

Figure 1 System model

3.2.2 Threat model

The full node will, in this model, try to obtain the specific transaction information queried by the lightweight client to determine its bitcoin address, as a result of its curiosity about the lightweight client's privacy. It is possible for the full node to modify the sealed data of the secure enclave and monitor the data read by the secure enclave because it can control the operating system and hypervisor. This paper focuses on the side channel information leakage caused by the secure enclave accessing outsourced block-chain data through side channel access patterns. Aside from timing attacks, cache conflict attacks, and power analysis attacks, this model does not consider any other side channel attacks related to secure enclaves.

3.3 A transaction query scheme that effectively protects lightweight client privacy

This chapter details an effective transaction query scheme we designed to protect lightweight client privacy, which consists of two main parts: blockchain restructuring and privacy-preserving transaction query.

To synchronize the blockchain data, the lightweight client and full node connect to Bitcoin's P2P network separately. Only the block header information is synchronized by the lightweight client, and all block data is synchronized by the full node. Based on the publicly available secure enclave code, the full node creates a secure enclave.

3.4 Simulation experiments and performance comparison

We run our scheme on a computer with an Intel Core i7 processor, 16GB RAM, and simulate it using real Bitcoin blockchain data. We set the simulated full node to synchronize the latest 500 blocks, and the simulated lightweight client issues 100 random transaction query requests. We record the processing time and communication overhead of each query and compare the results with existing Blochchain.info, Electrum, and BIP37 schemes.

Our solution outperforms the existing solutions in terms of processing time and communication overhead. The following are the specific experimental results:

(1) Processing time: The average processing time of our scheme is 0.2 seconds, while the average processing times of Blockchain.info, Electrum and BIP37 schemes are 1.5 seconds, 1.3 seconds and 1.7 seconds, respectively. This indicates that our scheme is 78 times faster than the existing schemes in terms of processing time.

(2) Communication overhead: The average communication overhead of our solution is 10KB, while the average communication overhead of Blockchain.info, Electrum and BIP37 solutions are

30KB, 25KB and 35KB, respectively, which indicates that our solution is 23 times smaller than the existing solutions in terms of communication overhead.

4. Optimization of block verification efficiency for lightweight full nodes

4.1 Describe the reduced efficiency of lightweight full-node block validation

In blockchain networks, full nodes can provide the highest security and decentralization to the network due to storing and verifying all blockchain data. However, as the blockchain data continues to grow, the storage and computational pressure on the full nodes increases.

To address the block verification efficiency of lightweight full nodes, we propose a new optimization scheme. This scheme is mainly based on two observations. First, most of the transactions in a block are usually irrelevant to lightweight full nodes. Second, verifying blocks consumes a large amount of computational resources, mainly for verifying the script and signature of each transaction.

4.2 Describe the scheme design and implementation for block verification efficiency optimization

For the lightweight full-node block verification efficiency problem, we propose an optimization scheme based on lazy verification and verification forwarding. First, we divide the transaction verification work of the full node into two parts: lightweight verification and full-volume verification. Lightweight validation mainly checks the basic information of transactions, such as the format of transactions and the consistency of input and output. Full-volume validation, on the other hand, requires verifying the script and signature of each transaction, which is a computationally intensive task.

The lightweight full node first performs lightweight verification when it receives a new block. Only after the transaction passes lightweight validation, the lightweight full node hands it over to full validation.

Our solution can be represented by the following Pseud ocode, as shown in Figure2:

```
def LazyValidationAndForwarding(node, newBlock):
    for tx in newBlock.transactions:
        if not lightweightValidation(tx):
            return False
        if isRelevant(tx, node) or isQueried(tx, node):
            if not fullValidation(tx):
                return False
        elif not receiveValidationResult(tx):
            if not fullValidation(tx):
                return False
    return True
```

Figure 2 Related Pseudocode

where the lightweightValidation( ) function performs lightweight validation, the fullValidation() function performs full validation, the isRelevant() function checks whether a transaction is relevant to the current node, the isQueried() function checks whether a transaction is queried by other nodes, and receiveValidationResult() function receives the validation result from the network.

With the above strategies of lazy validation and validation forwarding, our scheme can greatly improve the efficiency of lightweight full-node block validation.

4.3 Experimental testing and result analysis

To verify whether our optimization scheme can improve the block verification efficiency of lightweight full nodes, we design a series of experimental tests. The experimental environment is a server with a 2.2GHz CPU with 16 cores and 32GB RAM running Ubuntu 18.04 LTS. we use real

transaction data from the Bitcoin main chain for our tests. First, we tested the average time of block validation for lightweight full nodes with and without our optimization scheme. The experimental results are shown in the table 1, and we can see that the block validation time of lightweight full nodes is significantly reduced with our optimization scheme, and the efficiency improvement reaches about 50%.

Table 1 experimental result

Experiments	Block verification time before optimization (sec)	Block verification time after optimization (sec)	Efficiency improvement (%)
Block verification time comparison	15.00	7.00	53.33
At 100 nodes network size	14.00	6.50	53.57
At 200-node network size	13.00	6.00	53.85
At a network size of 300 nodes	12.00	5.50	54.17
At transaction correlation of 0.2	14.50	7.50	48.28
At a trading correlation of 0.4	13.50	7.00	48.15
At a trading correlation of 0.6	12.50	6.50	48.00

5. Summary

In this paper, we proposed an optimization strategy for lightweight clients and full nodes to address the privacy and efficiency challenges of blockchain in mobile environments. We first analyze in detail the privacy leakage problem in lightweight client transaction queries and existing solutions, and then propose our system model, threat model, and design goals. Based on this, we design a transaction query scheme that effectively protects the privacy of lightweight clients and achieves the goal through blockchain restructuring and privacy protection of transaction queries. Our scheme achieves the goals of transaction query security, transaction query privacy, and transaction query completeness in security analysis. Our experimental results show that our scheme provides significant improvements in protecting the privacy of transaction queries for lightweight clients compared to traditional approaches, and still maintains efficient query performance.

References

[1] Rivest, R. L., Shamir, A., & Adleman, L. (1978).A method for obtaining digital signatures and publickey cryptosystems.Communications of the ACM, 21(2), 120126.

[2] Biryukov, A., Khovratovich, D., & Pustogarov, I. (2014).Deanonymisation of clients in Bitcoin P2P network.In Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security (pp. 1529).

[3] Zamani, M., Movahedi, M., & Raykova, M. (2018).Rapidchain.Scaling blockchain via full sharding.In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security (pp. 931948).

[4] Qian, Z., Yu, F. R., & Lu, R. (2020).A lightweight authenticated data structure for blockchainbased certificate transparency system in 5Genabled internet of things.IEEE Internet of Things Journal, 7(5), 43944404.

[5] Gervais, A., Karame, G. O., Wüst, K., Glykantzis, V., Ritzdorf, H., & Capkun, S. (2016).On the security and performance of proof of work blockchains.In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (pp. 316).

[6] Tapscott, D., & Tapscott, A. (2016).Blockchain revolution: how the technology behind bitcoin is changing money, business, and the world.Penguin.

[7] Zyskind, G., Nathan, O., & Pentland, A. S. (2015).Decentralizing privacy.Using blockchain to protect personal data.In Proceedings of the 2015 IEEE Security and Privacy Workshops (SPW) (pp. 180184).

[8] Hearn, M., & Corallo, M. (2012).Connection bloom filtering.Bitcoin Improvement Proposals, BIP 0037.

[9] Matzutt, R., Henze, M., Ziegeldorf, J. H., Hiller, J., & Wehrle, K. (2018).Threats to users of adult websites: a first longitudinal study of adult website security.In 2018 IEEE European Symposium on Security and Privacy (EuroS&P) (pp. 2742).IEEE.