University of Hong Kong

Final Year Project

Augmenting Blockchains with A Faster Peer-to-Peer Network Protocol

Project Plan

Haoran Qiu and *Tao Ji*

Department of Computer Science

Supervised by

Dr. Heming Cui

Department of Computer Science

September 30, 2018

Contents

1 Introduction

A blockchain is essentially a distributed ledger that permanently records the transactions between two parties [1]. The transactions recorded are verifiable and resistant to modification. This feature of security has contributed to the emergence of cryptocurrencies that leverage the blockchain as their cornerstone, the most salient example being Bitcoin [2]. Despite the prosperity of cryptocurrencies, general and all-purpose blockchains that accommodate various applications, such as Ethereum [3] have been proposed. However, although Ethereum is a Turing-complete system [3], at the essence it is designed for the cryptocurrency based on it. So the Proof-of-Work (PoW) consensus is used to support the valuation of the cryptocurrency in the socioeconomic sense, which results in poor efficiency. To facilitate general-purpose applications in a more efficient, other consensus protocols have been proposed to improve the performance of the blockchain in different scenarios (See Section 2.1).

The emergence of trusted execution environments (TEE) has enabled people to eliminate the assumption of adversarial nodes in a distributed system. Under such context, some more efficient blockchain-based distributed computing systems along with the underlying consensus protocols are proposed, for example, the Proof-of-Elapsed Time (PoET) consensus under Hyperledger Sawtooth [4]. While much effort has been devoted to the development of new consensus protocols, one aspect where fewer works have explored is the peer-to-peer (P2P) network beneath the blockchain system. The P2P protocol is not a novel field, but it has been shown that it can be enhanced with TEE [5].

One of the techniques that might enable further improvements of the existing P2P protocols under blockchains is network function virtualization (NFV). With NFV, upper-layer applications are allowed to control the lower-layer functionalities such as routing. One of the problems of the existing P2P protocols is the bandwidth consumption caused by redundant messages, which is a waste of resource. One possible way to optimize the resource usage can be to choose a few trusted (TEE-guarded) nodes as privileged ones to oversee the messages in the network and design an algorithm for them to collaborate so as to reduce redundancy. This is merely a tentative idea for introductory purposes. Details of the objective are discussed in Section 3.

The following of this report is structured in this way: Section 2 provides some background of consensus protocols, P2P networks, and TEE; Section 3 presents the initial objective of

this project; Section 4 proposes the methodology to be applied in this project; Section 5 discusses several challenges we expect and Section 6 shows the detailed project schedule.

2 Background

This section provides the background of three significant terms mentioned in Section 1.

2.1 Consensus Protocol

Consensus protocols are crucial for distributed systems. Classical ones include Two-Phase Commit and Practical Byzantine Fault Tolerance (PBFT) [6], which are designed to target various failure models. The emergence of Proof-of-Work (PoW) used in Bitcoin to achieve consensus in a fully decentralized public network motivated a new class of consensus protocols based on Proof-of-X, such as Proof-of-Stake [7] and Proof-of-Luck [8]. Some other protocols designed for permissioned blockchains might employ TEE to improve the efficiency. For example, Hyperledger Sawtooth [4] utilizes Intel Software Guard eXtentions (SGX, introduced below) to trust nodes and proposes Proof-of-Elapsed Time believed to be highly efficient, yet scalibility might not be a critical performance in this consensus.

GEEC is a new public blockchain protocol which leverages the strong confidentiality and integrity of the hardware [9]. The key insight of this work is that existing protocols rely on either computation power like PoW or possessed coins such as PoS since there is no trust established among each other. GEEC makes use of Intel SGX to guarantee the integrity of code and execution so that efficiency and consistency can be ensured at the same time. In addition to the features common with PoET, this blockchain is also said to provide high scalibility.

2.2 P2P Network

A blockchain is built on top of a decentralized peer-to-peer(P2P) network used to propagate system information such as transactions or chain member updates [10]. Besides its decentralization feature, the strength of P2P network also includes self-organization, load-balancing, adaptation, and fault-tolerance. Though there are many benefits P2P network

can provide to its atop applications like blockchains and Bittorent, the communication on a P2P network suffers from message redundancy [11, 12].

There are two approaches to perform the broadcast operation: the flooding approach and the distributed hash table approach [13]. To reduce as many messages as possible, there should be less repeated messages sent to the same node. Meanwhile, the property required by the blockchain application on top of the network should not be affected. With Trusted Execution Environment (TEE) on Intel SGX, some key protocol components like routing algorithm and network information can be put in the hardware enclave [14, 15].

2.3 Trusted Execution Environment

Trusted computing was defined to help systems to achieve secure computation, privacy and data protection. Originally, the trusted platform module (TPM) allows a system to provide evidence of its integrity in a separate hardware module. In recent years, a new approach to address trusted computing appears which allow the execution of arbitrary code within a confined environment that provides tamper-resistant execution to its applications - trusted execution environment (TEE) [16]. TEE is a secure, integrity-protected processing environment, consisting memory and storage capabilities [17].

Intel SGX is one popular instance of TEE which is a set of extensions to the Intel architecture that aims to provide integrity and confidentiality guarantees to security sensitive computation performed on a computer where all the privileged software (kernel, hypervisor, etc.) is potentialy malicious [14]. Intel SGX provides two kinds of attestations(local and remote) to prove that particular piece of code is running in a genuine SGX-enabled CPU [9] and also provides a trustworthy source of random number [18]. Currently there is a related work which uses Intel SGX to provide reliable broadcast for P2P network [5]. However, there is no related work on using Intel SGX to improve asynchronous P2P network performance, which is the main focus of this project.

3 Objective

3.1 Scope

Considering the complexity of conducting this research, the scope is temporarily decided to the following and is subject to changes in the actual research progress:

Asynchronous P2P Protocol. There are two types of peer-to-peer networks: synchronous networks and asynchronous networks[19]. The main topic researched in this project is to design an asynchronous P2P protocol that achieves higher efficiency leveraging Intel SGX and NFV, instead of a synchronous protocol, since in most blockchains, synchronization is the responsibility of the consensus protocol. In addition, this protocol should not sacrifice any property in the original protocol required by blockchains.

System Implementation. After the P2P protocol is finished, we implement a blockchain system by replacing the P2P layer of existing systems with our P2P protocol. The reasons not to develop a brand new system are: i) the time constraint of this project could be too tight to build from scratch; ii) we would like to find the performance gain of the system where the only part changed is the P2P protocol to illustrate our contribution.

A public blockchain system will most likely be implemented. Public blockchains and permissioned blockchains are mostly similar to each other in three ways: i) they are both based on the decentralized peer-to-peer network; ii) they both maintain replicas in sync through a consensus protocol; iii) they provide certain guarantees on the immutability of the ledger [20]. The reason for us to choose public blockchains is that they are open and more challenges are incurred.

Evaluation. Finally, we evaluate the protocol by comparing the performance of our system and the original one (before replacing the P2P protocol).

3.2 Deliverable

This project will deliver: i) a new asynchronous protocol for P2P network with improved performance; ii) a blockchain system, implemented by replacing the P2P protocol of some existing blockchain systems like GEEC [9] and ByzCoin [21]; iii) an evaluation of the newly implemented system.

The project progress can be checked on the website: https://i.cs.hku.hk/∼ fyp18006

4 Methodology

In order to make sure the project is delivered successfully, we divided the project into three main phases and the methodology for each phase is:

Protocol Design. We will find what researchers were doing on improving the performance of peer-to-peer network and figure our what is preventing them from making it more efficient. After having done literature review about peer-to-peer network and Intel SGX, we can think about how to make use of Intel SGX to mitigate the problem unsolved.

System Implementation. We will first do some research on existing blockchain systems such as GEEC and extract the peer-to-peer network layer from them. Modifying the network layer and then integrate it with the original system.

Evaluation. After having the system working, we are going to evaluate our system compared to other existing blockchain systems running the same application. The metrics for performance tentatively include but are not limited to: i) message throughput of the same operation; and ii) maximum operation throughput on the same network.

5 Challenges and Mitigation

We are expecting two main challenges in this project:

- To improve the efficiency of the P2P protocol with all other properties required by blockchain kept
  We will first design a naive protocol which satisfies the requirement and then think about ways to improve it under various possible attacks. Since Intel SGX provides guarantee on code and execution integrity, routing algorithms and some network information can be put in the hardware enclave. We can start from what Intel SGX can do to design the initial protocol.
- To implement the system with NFV and Intel SGX
  We plan to do some research on securing NFV with SGX like in [22] and study

programming skills on SGX.

6 Project Schedule

Time Periods	Tasks
September	• Meeting with the supervisor • Project plan writing • Project website designing • Literature review
October	• Literature review • Scratch and discuss the first version of the protocol
November - December	• Complete the protocol design • Start to implement the system • Interim report writing
January - February	• System Implementation
March - April	• System evaluation • Final report writing • Final presentation • Poster design • Exhibition

7 Conclusion

This is a detailed plan for our project. Bitcoin and other cryptocurrency applications have thrived because of their decentralization nature and trust model. While much effort has been made in the consensus protocols, little research is done in augmenting the underlying P2P protocols using state-of-the-art techniques such as TEE and NFV. Therefore, in this

project, we will try to design such a protocol and implement it in an existing blockchain system. We will also evaluate and discuss our protocol and the system. Hopefully some improvements can be achieved.

The project website is: i.cs.hku.hk/∼ fyp18006.

8 References

[1] Marco Iansiti and Karim R. Lakhani. *The Truth About Blockchain*. URL: https://hbr.org/2017/01/the-truth-about-blockchain (visited on 30/09/2018).

[2] Satoshi Nakamoto. 'Bitcoin: A peer-to-peer electronic cash system'. In: (2008).

[3] Gavin Wood. 'Ethereum: A secure decentralised generalised transaction ledger'. In: *Ethereum project yellow paper* 151 (2014), pp. 1–32.

[4] Intel Corporation. *Introduction*. URL: https://sawtooth.hyperledger.org/docs/core/releases/latest/introduction.html (visited on 30/09/2018).

[5] Yaoqi Jia et al. 'Robust Synchronous P2P Primitives Using SGX Enclaves.' In: *IACR Cryptology ePrint Archive* 2017 (2017), p. 180.

[6] Miguel Castro, Barbara Liskov et al. 'Practical Byzantine fault tolerance'. In: *OSDI*. Vol. 99. 1999, pp. 173–186.

[7] Aggelos Kiayias et al. 'Ouroboros: A provably secure proof-of-stake blockchain protocol'. In: *Annual International Cryptology Conference*. Springer. 2017, pp. 357–388.

[8] Mitar Milutinovic et al. 'Proof of luck: An efficient blockchain consensus protocol'. In: *Proceedings of the 1st Workshop on System Software for Trusted Execution*. ACM. 2016, p. 2.

[9] X. Chen et al. 'GEEC: Scalable, Efficient, and Consistent Consensus for Blockchains'. In: *ArXiv e-prints* (Aug. 2018). arXiv: 1808.02252 [cs.DC].

[10] Joan Antoni Donet Donet, Cristina Pérez-Sola and Jordi Herrera-Joancomartı. 'The bitcoin P2P network'. In: *International Conference on Financial Cryptography and Data Security*. Springer. 2014, pp. 87–102.

[11] Hassan Barjini and Mohmed Othman. 'SmoothFlood: Decreasing redundant messages and increasing search quality of service in peer-to-peer networks'. In: *Information Retrieval & Knowledge Management,(CAMP), 2010 International Conference on*. IEEE. 2010, pp. 138–142.

[12] L. Guo et al. 'LightFlood: Minimizing Redundant Messages and Maximizing Scope of Peer-to-Peer Search'. In: *IEEE Transactions on Parallel & Distributed Systems* 19 (Sept. 2007), pp. 601–614. ISSN: 1045-9219. DOI: 10.1109/TPDS.2007.70772. URL: doi.ieeecomputersociety.org/10.1109/TPDS.2007.70772.

[13] Sameh El-Ansary et al. 'Efficient broadcast in structured P2P networks'. In: *International workshop on Peer-to-Peer systems*. Springer. 2003, pp. 304–314.

[14] Victor Costan and Srinivas Devadas. 'Intel SGX Explained.' In: *IACR Cryptology ePrint Archive* 2016.086 (2016), pp. 1–118.

[15] Frank McKeen et al. 'Intel® software guard extensions (intel® sgx) support for dynamic memory management inside an enclave'. In: *Proceedings of the Hardware and Architectural Support for Security and Privacy 2016*. ACM. 2016, p. 10.

[16] Mohamed Sabt, Mohammed Achemlal and Abdelmadjid Bouabdallah. 'Trusted execution environment: what it is, and what it is not'. In: *14th IEEE International Conference on Trust, Security and Privacy in Computing and Communications*. 2015.

[17] N Asokan et al. 'Mobile trusted computing'. In: *Proceedings of the IEEE* 102.8 (2014), pp. 1189–1206.

[18] *Software Guard Extensions Programming Reference*. http://kib.kiev.ua/x86docs/SDMs/329298-001.pdf. [Online; accessed 30-September-2018].

[19] Sonja Buchegger et al. 'PeerSoN: P2P social networking: early experiences and insights'. In: *Proceedings of the Second ACM EuroSys Workshop on Social Network Systems*. ACM. 2009, pp. 46–52.

[20] Garry Gabison. 'Policy considerations for the blockchain technology public and private applications'. In: *SMU Sci. & Tech. L. Rev.* 19 (2016), p. 327.

[21] Eleftherios Kokoris Kogias et al. 'Enhancing bitcoin security and performance with strong consistency via collective signing'. In: *25th USENIX Security Symposium (USENIX Security 16)*. 2016, pp. 279–296.

[22] Ming-WeiShih MohanKumar TaesooKim AdaGavrilovska. 'S-NFV: Securing NFV states by using SGX'. In: (2016).