#6 A Two-Layer Blockchain Sharding Protocol Leveraging Safety and Liveness for Enhanced Performance

Email notification
Select to receive email on updates to reviews and comments.

PC conflicts
None

Accepted

Submission (1MB) 28 Jun 2023 7:50:22pm AoE · 6b2d4015

Abstract

Sharding is a critical technique that enhances the scalability of blockchain technology. However, existing protocols often assume the adversarial nodes in a general term without considering the different types of attacks, which limits transaction throughput in runtime because the attack on liveness could be mitigated. This paper introduces Reticulum, a novel sharding protocol that overcomes this limitation and achieves enhanced scalability in a blockchain network.

Reticulum employs a two-phase design that dynamically adjusts the transaction throughput based on the runtime adversarial attacking either or both liveness and safety. It consists of ``control" and ``process" shards in two layers corresponding to the two phases. Process shards are subsets of control shards, with each process shard expected to contain at least one honest node with high confidence. Conversely, control shards are expected to have a majority of honest nodes with high confidence. In the first phase, transactions are written to blocks, which are then voted on by nodes in the corresponding process shards. Blocks that receive unanimous acceptance verdicts are accepted. In the second phase, blocks that do not obtain unanimous verdicts are collected and voted on by the control shards. A block is accepted if the majority of nodes vote to accept it. The opponents and silent voters of the first phase will be eliminated. In summary, Reticulum leverages unanimous voting in the first phase to involve fewer nodes for accepting/rejecting a block, allowing more parallel process shards. The control shard comes into play as a liveness rescue when disputes arise.

Experiments demonstrate that the unique design of Reticulum empowers high transaction throughput and robustness in the face of different types of attacks in the network, making it superior to existing sharding protocols for blockchain networks.

Authors (anonymous)
Yibin Xu (University of Copenhagen) <yx@di.ku.dk>
Jingyi Zheng (University of Copenhagen) <jrb385@alumni.ku.dk>
Boris Düdder (University of Copenhagen) <boris.d@di.ku.dk>
Yongluan Zhou (University of Copenhagen) <zhou@di.ku.dk>
Tijs Slaats (University of Copenhagen) <slaats@di.ku.dk>

✓ **Intention to submit artifact**

Topic
Security and privacy for blockchains and cryptocurrencies

	OveRec	WriQua	RevExp	Eth
Review #6A	3	4	1	3
Review #6B	3	3	3	3
Review #6C	4	3	1	3

1 Comment: *Rebuttal Response* (Y. Xu)

You are an **author** of this submission.

Edit submission Add comment

Reviews and comments in plain text

Review #6A

Overall recommendation

3. Weak accept

Writing quality

4. Well-written

Reviewer expertise

1. No familiarity

Paper summary

Blockchain sharding is a method that aims to improve the scalability of a vote-based blockchain system by randomly dividing the network into smaller divisions, called shards. The idea is to increase parallelism and reduce the overhead in the consensus process. However, dividing the network into small subsets (shards) makes the voting process more vulnerable to corruption. In this paper, it is proposed Reticulum, a new secure protocol which introduces low overhead and it is able to inhibit adversarial behaviors. The performance of Reticulum has been compared to the state-of-the-art. Moreover, a prototype of the protocol was developed and published in open source.

Strengths

- The new protocol outperforms the state-of-the-art
- A prototype of the protocol is published in open source and will provide researchers with a valuable resource for studying blockchain sharding

Weaknesses

- Lack of experimental comparison with the state-of-the-art

Detailed comments for authors

The work is solid. The new protocol is described and evaluated thoroughly. Reticulum is more secure and inserts less overhead than previous solutions. It is also able to remove nodes considered malicious from the network.

The only weakness regards the comparison with the state-of-the-art, which is not completely fair, since Gearbox has been simulated only mathematically due to the lack of source code.

Finally, it is not very clear how the proposed solution could be integrated into the proof-of-stake protocol.

Ethics

3. No ethic issues

Review #6B

Overall recommendation

3. Weak accept

Writing quality

3. Adequate

Reviewer expertise

3. Knowledgeable

Paper summary

The paper presents Reticulum, a two-layer sharding protocol leveraging safety and liveness. The first layer focuses on safety, where process shards are set to be as small as possible to utilize parallelism, at the cost of the lowest tolerance to liveness attack. In the second layer, shards come into play for consensus finalization and ensure the liveness of the entire sharding system. The authors also provided experimental results for Reticulum in terms of throughput under different attack settings.

Strengths

- Timely and interesting topic
- Clear contribution
- Interesting mechanisms to utilize parallel shards without compromising liveness with a control layer in a sharding system.
- Detailed mathematical analysis
- Good experimental evaluation

Weaknesses

- No experimental comparison with previous protocols
- The presentation of the protocol can be improved and somehow hard to follow

Detailed comments for authors

- Reticulum decouples the block unanimous voting and finalization phases in blockchain sharding protocol to leverage safety and liveness. The protocol seems sound to me. However, the presentation of the protocol should be rechecked (or improved).

- In algorithm 2, line 5 and line 9, blocks and votes are broadcast to nodes in the same control shard. As is the first phase protocol, should these messages be sent to nodes in the process shard and the corresponding control shard? Or the "node" running this procedure is a node in a control shard? Similarly, in the second phase protocol, there's no explanation for the identity of the node that running the protocol.

Besides, it seems that the term "node" in a protocol has several different meanings, i.e., a process node, a control node, or a parameter used for enumeration, which is confusing.

- The output of the first phase protocol is a symbol of "ACCEPTED" or "FAILED". However, the trigger conditions of the second phase protocol aren't relevant to the symbol. According to the description, the second phase of the protocol could be triggered with an expired timer and uses messages sent from process nodes as input. In algorithm 2, the votes in the first phase of the protocol are sent to control nodes using a BB protocol. While the description of a BB protocol is only provided in the following section in an informal approach for complexity analyzing. Clear codes for the exact trigger condition and timer setting for the second phase protocol should be added and discussed.

- No experimental results are conducted for protocols other than Reticulum. Some comparison of related works seems to be necessary.

- Typos: Page 4, right, "Λ = bβ/Ncc", this is inconsistent with the description on the left, should this be "Λ = bN/Ncc"? Page 7, Algorithm 3, line 6, "VerifyProcessBlock(B)", inconsistent to that on line 17. Page 10, left, "replicae" Page 12, right, "Gearbox recommend the usage of four gears gears...",

Concrete steps for improvement

- improve the organization of the paper
- clarify the details

Ethics

3. No ethic issues

Review #6C

Overall recommendation

4. Accept

Writing quality

3. Adequate

Reviewer expertise

1. No familiarity

Paper summary

This work proposes a new sharing protocol, which separately considers attacks on liveness and safety and providing methods for liveness attack inhibition. Such design can improve the performance of consensus.

Strengths

1. A new sharing protocol is proposed.
2. The security of the proposed approach is discussed.

Weaknesses

1. This paper claims that the prototype has been open source, but the source code is not provided to the reviewers.

Detailed comments for authors

/

Concrete steps for improvement

/

Ethics

3. No ethic issues

Rebuttal Response

Author [Yibin Xu] 5 Sep 2023 690 words

We appreciate the reviewers' insightful assessments of our paper. We are committed to addressing the comments and suggestions by revising the paper. Nonetheless, we would like to clarify the following points:

Reviewer A:

1. Regarding experimental comparison with state-of-the-art solutions:

   a) We have indeed experimentically compared Reticulum with our implementation of RapidChain, referred to as the baseline. This implementation encompasses the entire shard generation and block finalization of RapidChain, providing a suitable foundation for assessing the performances of various technical designs discussed in our paper.

   b) The contribution of the Gearbox paper is to show that one may leverage liveness and safety threshold to adjust the shard size. The paper only provided a high-level Universal Composability framework but did not provide comprehensive design details. The authors specifically stated in their paper that they “only formalize the features relevant for our sharding protocol and abstract away other details.” Without the design details, we are unable to implement Gearbox. In addition, the Gearbox paper did not present any experimental results nor claim that Gearbox was implemented. Therefore, it is reasonable to assume that Gearbox has never been implemented, and it is unknown how it can be implemented.

   c) To provide a fair comparison, we set up our experiments in a way that favors Gearbox. We simulated the optimal theoretical performance of Gearbox without considering the runtime costs of Gearbox, including network delays and shard rebuilding overhead. On the other hand, all the runtime overheads are represented in the results of Reticulum obtained via experiments using our prototype.

   All in all, our simulation of Gearbox is the only viable method for us to establish a fair comparison.

   d) Finally, our paper highlights significant security vulnerabilities in Gearbox, overshadowing its performance.

   We will further justify our experimental setup in the revised paper.

2. Regarding how Reticulum can be integrated into the proof-of-stake protocol:

   There are two schools of proof-of-stake protocols. The first uses the longest-chain mechanism to reach consensus, where a block is finalized when it is in the longest-chain and has a pre-defined generation of descendants. The other uses voting for block finalization. Reticulum can be plugged into any voting-based consensus protocol. For example, Casper FFG is a vote-based block finalization method adopted by Ethereum’s Gasper consensus protocol. Therefore, Reticulum can be seamlessly integrated into Ethereum. We will clarify this in the revised paper.

Reviewer B:

1. Regarding the witness of no experimental comparison:

   We refer the reviewer to our response to the first point of reviewer A.

2. Regarding the typos and other questions.

   a) The reference to "nodes in the same control shard" in algorithm 2, lines 5 and 9, was indeed a typo. The correct reference is to "nodes in the same process shard."

b) In the context of the second-phase protocol, the term "node" pertains to all nodes within each control shard, and consequently, each node executes the second-phase protocol because each node belongs to one of the control shards. To clarify, we will rename the nodes into process and control nodes depending on their roles in the two-phase protocol.

c) Regarding the discrepancy in the output of the first-phase protocol and the trigger conditions of the second-phase protocol, we concur with the reviewer's observation. The second phase happens regardless of the output of the first phase. So when the time bound for the first phase passes, the second phase is triggered. However, only the failed first-phase blocks will be synced in the second phase. Therefore, the number of failed first-phase blocks affects the data and time needed for the second phase. In the revised version, we will explicitly address the transition between the first and second phases, clarifying that the second phase is initiated independently of the first-phase outcome.

d) Regarding the other typos, we will correct them and proofread the paper more carefully. We appreciate the reviewer for pointing out the identified typos.

Reviewer C:

1. Regarding the concern about the open-source prototype, we were not allowed to submit it as an attachment in the submission system at the time of paper submission. If the paper is accepted, we intend to submit it for the Artifact Evaluation.