TL;DR

In L2 blockchains, sequencers and proposers play vital roles. Sequencers validate transactions and bundle them for L1 transmission, while proposers submit state commitments to L1. Failures in these entities can occur, leading to censorship or asset locking issues but barely escape hatches work on any L2, this topic introduces solution to solve the same.

Current Scenario

As of the current scenario, the L2 blockchain ecosystem faces potential challenges related to sequencer and proposer failures. Sequencers, responsible for validating and bundling transactions for L1 transmission, may introduce risks such as censorship or shutdown. Similarly, proposers, tasked with submitting state commitments to L1, could face failures leading to asset locking issues.

These challenges are exacerbated by the fact that sequencers and proposers are primarily operated by centralized entities, introducing additional risks such as MEV exploitation or sequencer shutdown. The crucial aspect is not only the use of decentralised sequencers, but also the successful implementation of fraud and validity proofs, as well as the inclusion of an escape hatch mechanism.

To tackle sequencer failures, solutions such as “Self Sequencing” and “Enque via L1” have been introduced. Similarly, for proposer failures, mechanisms like the “Escape Hatch” and “Self Propose” are available. Unfortunately, escape hatches and self-sequencing were not adequately prioritized in the design and implementation, representing a notable security vulnerability. Additionally, many L2 solutions currently lack these critical features, leaving a gap in addressing potential failures.

The information provided by L2beat indicates a lack of robust implementation in the mechanisms designed to address sequencer and proposer failures.

[

Screenshot 2023-11-20 at 8.14.30 PM

1542×1198 147 KB

](https://global.discourse-cdn.com/standard17/uploads/arbitrum/original/2X/3/3d322675716cd154bcb605e68c0dc319fa27b595.png)

[

Screenshot 2023-11-20 at 8.12.12 PM

1542×1262 160 KB

](https://global.discourse-cdn.com/standard17/uploads/arbitrum/original/2X/3/3dee761d999ed514bb8f583706f4fa8abb3bae55.png)

How L2s can implement self sequence and escape hatch mechanisms

Self sequencing

- A backup strategy permits transactions to be forcefully included in the case of sequencer failure by routing them through L1 after a specified time interval has expired. This approach ensures that transactions that may have been delayed or impacted by the L2 sequencer failure can still be processed and validated on the principal blockchain, preserving the overall transaction processing system’s integrity and continuity.
- Implementing a time interval is a technical safety meant to establish a temporal buffer before the forced inclusion of transactions via Layer 1 in response to sequencer failure. This temporal delay provides a controlled window for identifying and resolving potential issues, performing appropriate system checks, and restoring operational stability.
- When a user wants to force the inclusion of a transaction, a specialised function must be called on L1 rollup contract which then relays the transaction to the delayed queue of L2 sequencer. This delayed queue has two primary purposes: first, as a staging place for transactions awaiting processing, and second, as a security precaution to prevent potential malicious actors from participating in unwarranted self-sequencing attempts. Before being included in the delayed queue, transactions must be scrutinized and subjected to predetermined criteria. This protects against unauthorized or malicious efforts to modify the sequencing process, ensuring the L2 environment’s integrity and security.
- Following a set time threshold, transactions in the delayed queue are subjected to a batch inclusion process, which includes a number of mandatory checks. This temporal constraint serves as a regulated tool for regulating transaction processing timing. Transactions gathered in the delayed queue during this period are pooled together into a batch, and the inclusion procedure begins, assuming that preset validation tests are passed successfully. The technical

complexity comes from organising this batch inclusion, in which the system guarantees that transactions match established criteria before being integrated into the L2.

[

Screenshot 2023-11-20 at 9.50.25 PM

1622×850 30.3 KB

](https://global.discourse-cdn.com/standard17/uploads/arbitrum/original/2X/5/5c9291787c6341f5c4c89e09baac55292ab8f94e.png)

Escape Hatches

- The proposer is needed to send a predetermined message to the smart contract on a regular basis. This message is a preventative precaution designed to ensure the proposer's continuous operational integrity. The absence of this communication within a predetermined time range causes the proposer to be declared non-operational or defective.
- In the case that the proposer's operations halt or that the designated message is not transmitted, an automatic mechanism must begin to execute the freezing of assets on L2. The freezing procedure should comprise of secure locking of assets within a specific smart contract issued on L2. This step effectively prevents any subsequent transactions involving the frozen assets from taking place until the underlying issue with the proposer is resolved. This upgrade assures that any attempts to transfer or modify the frozen assets are methodically stopped, reducing possible dangers connected with the malfunctioning proposer.
- Users can trigger escape hatch mechanism which developed to allow users to reclaim ownership of their assets under certain scenarios, most notably when the proposer is confirmed to be inactive. This mechanism is designed to be used only when stated criteria conclusively identify the proposer's non-functionality.
- In order to reclaim ownership of their assets, users are required to generate Merkle proofs substantiating their fund holdings. These Merkle proofs serve as cryptographic proof of the specific assets owned by users when they are locked. Users then submit these Merkle proofs to the L1 contract, which then goes through a verification procedure to authenticate the legitimacy and ownership of the proofs presented. Only after the proofs have passed this stringent validation is the Layer 1 contract authorised to begin transferring the locked assets to the users' control.

[

Screenshot 2023-11-20 at 10.27.03 PM

1312×320 6.6 KB

](https://global.discourse-cdn.com/standard17/uploads/arbitrum/original/2X/7/7135f837a404a399b61bb759164785dd39cd717c.png)

[

Screenshot 2023-11-20 at 10.25.22 PM

926×548 15.3 KB

](https://global.discourse-cdn.com/standard17/uploads/arbitrum/original/2X/a/adfd36bf9f87cc724aa541df121118dc8cc8b872.png)

Conclusion

The analysis emphasises the vital relevance of overcoming sequencer and proposer failures in L2 blockchains to ensure transaction processing robustness and security. Current issues and weaknesses, including the lack of prioritisation in escape hatches and self-sequencing, emphasise the necessity for comprehensive approaches throughout L2 ecosystems. Implementing these solutions can improve the overall dependability and robustness of L2 blockchain systems, giving consumers more trust in the integrity of their transactions and assets.