Arbitrum GovHack Track

:

Failover/Backup Sequencers

Challenge Statement:

- Downtime: This leads to a poor user experience for dApp builders and Arbitrum users, as well as loss of revenue for the DAO.
- Censorship: Since the sequencer is run by only one entity, that entity holds a monopoly and has the power to reject user transactions. This could also lead to a poor user experience for the users.

Members:

Sam from Node Guardians and Ben from Coinbase.

Key Terms:

Liveness (uptime):

-the Arbitrum One chain keeps processing user transactions

-censorship resistance i.e new honest transactions gets included in Arbitrum

Safety:

-reorg resistance i.e confirmed transactions stay confirmed

-data availability i.e Arbitrum data remains available

-validity i.e the transactions posted to Ethereum are valid.

Security: Liveness + Safety

Viability:

The reason we are working on this proposal is because it could serve as a purely additive security measure to the existing Arbitrum protocol—the safety of Arbitrum users' funds doesn't rely on the mechanism proposed. Its main purpose is to provide a fallback option in the event of unexpected situations such as downtime or censorship involving the Arbitrum sequencer.

Feasibility:

It's important to note that we are still investigating the exact costs of running sequencers in parallel, as we are reviewing the specific documentation that Offchain Labs shared with us.

The great thing is that there is no need to modify any code on Arbitrum, as it already supports the escape hatch (force inclusion mechanism). This allows users to not only escape but also enables the DAO to elect a new sequencer. The goal is to run the Arbitrum sequencer in parallel to be ready to serve the DAO should the need arise.

Desirability:

It's not news to anyone that security is crucial. Currently, the Arbitrum sequencer, which is responsible for collecting and ordering transactions, is operated by a single entity. Security includes both safety and liveness. From a safety perspective, the fact that the sequencer is run by only one entity does not compromise the protection of user funds, as the integrity of Arbitrum's state transitions are guaranteed by fraud proofs. However, security also covers liveness—the ability of the Arbitrum chain to continue processing transactions. Despite our best efforts, servers can fail, and the sequencer might censor transactions. This can lead to a poor user experience for both Arbitrum users and dApps by causing delays of several hours in the processing of their transactions.

Impact:

This approach would allow the DAO to start by electing 2 backup sequencers (Coinbase and Node Guardians) in case of an issue with the current Arbitrum sequencer. We believe it is crucial that this process occurs through governance rather than

through any form of $ARB staking mechanism. In reality, it doesn't seem to be relevant to give up a) value and b) the governance process on the operator side to another liquid staking protocol, specially when there's an opinionated and active governance in place, which we believe is the case for Arbitrum.

These backup sequencers would run in parallel, ready to replace the active one if anything goes wrong. Forexample, we have already witnessed an event where the Arbitrum sequencer went down for a few hours, preventing users from accessing the chain and the dApps deployed on Arbitrum One. Even though this event was more related to implementation details, our goal is to prevent such situations from recurring.

We aim to prioritize practical liveness/censorship resistance and having backup sequencers contribute to it. Having downtime or being censored, even for a short period, may impose a real cost on the user, depending on the situation (e.g time=money). Users generally desire 'real-time' censorship resistance and liveness at the speed of the blockchain they're using. Note that liveness failures, such as those Solana has experienced, are unacceptable for high-value DeFi, which continues to grow on Arbitrum.

Final thoughts:

We are aware that this is a complex technical topic, and we do not want to rush things or take shortcuts. What we have written here serves as an introduction, and we wish to take the time to consult with delegates before proceeding with an official proposal.

Loom:

Video recording software