

Constitutional

Abstract

This Constitutional AIP proposes two improvements to batch posting for Arbitrum One and Arbitrum Nova:

- Batch Poster Manager:

Introduce a “batch poster manager” role with the ability to grant/revoke batch-posting affordances.

- Increase MaxTimeVariation

: Change the max-time-variation “future blocks” and “future seconds” values to 64 and 768, respectively, in line with Ethereum’s proof of stake finality guarantees.

These changes will allow the system to be more resilient, and don’t represent a change to the system’s current trust model.

Motivation

Batch Poster Manager:

Currently, both Arbitrum One and Arbitrum Nova, each have a single address that is granted the batch-poster role (this is currently the same as the Sequencer). The Sequencer posts batches frequently, and thus the batch-poster address must be controlled by a hot wallet. This means that if the batch poster’s keys were compromised, Sequencing could be unstable until the DAO took action.

This AIP proposes a system in which a “batch poster manager” role is granted to the operator of the Sequencer which has the ability to grant and revoke batch-posting affordances. This way, the batch poster manager could perform key rotations for the batch posters— routinely, and/or if a batch poster address is ever compromised — quickly and without the DAO needing to take coordinated action. Note that this proposal does not change the sequencer, but more so allow for easier key rotations on the batch poster.

Crucially, this would not represent a change on the current system’s trust model:

- In both the current and the new proposed system, the Sequencer is entrusted with managing the batch posting affordance; in the current system, for example, the entity behind Sequencer could technically grant batch posting to an additional entity by simply sharing it’s keys.
- In the new system, the DAO would still

have the same ability to revoke the Sequencer role; i.e., the DAO could update the batch poster manager (along with any batch posters).

MaxTimeVariation:

The futureBlocks value in the the SequencerInbox enforces a max block height that a batch can be posted relative to the current block (likewise with futureSeconds). The current value for futureBlocks is 12, which was set prior to the Ethereum merge. A small value for future blocks means that a relatively small L1 reorg can cause an otherwise valid batch to revert. This proposal increases the value to 64, two epochs, in line with Ethereum’s finality guarantees.

Implementation

Batch Poster Manager:

github.com/OffchainLabs/nitro-contracts

[Merge pull request #74 from OffchainLabs/batch-poster-manager](#)

committed 03:55PM - 24 Jan 24 UTC

[

gzeoneth
(<https://github.com/gzeoneth>)

[+172

](https://github.com/OffchainLabs/nitro-contracts/commit/c7554852a7c41ca5eaef298dab10472a7f550df7)

Add a batch poster manager for the sequencer inbox

Note that this implementation is currently under audit and is dependent on the ArbOS 20 AIP changes [AIP: ArbOS Version 20](#)). Depending on the timeline of the audits, the result the ArbOS20 AIP acceptance, and the feedback on this proposal, these changes can be bundled into the same proposal as the ArbOS 20 changes or proposed separately.

MaxTimeVariation

:

github.com/ArbitrumFoundation/governance

[set sequencer inbox maxtimevariation action contract](#)

ArbitrumFoundation:main

← ArbitrumFoundation:seq-maxtimevariation-action

opened 07:32PM - 27 Dec 23 UTC

[

DZGoldman
](https://github.com/DZGoldman)

[+111

-0

](https://github.com/ArbitrumFoundation/governance/pull/233/files)

Audit: [publications/reviews/2024-01-offchainarbitrum-securityreview.pdf at master · trailofbits/publications · GitHub](#)