Obinna Nnodim

## Immuna Write Up

# Crypto Bridge Monitoring

*Photo by Luke Besley on Unsplash*

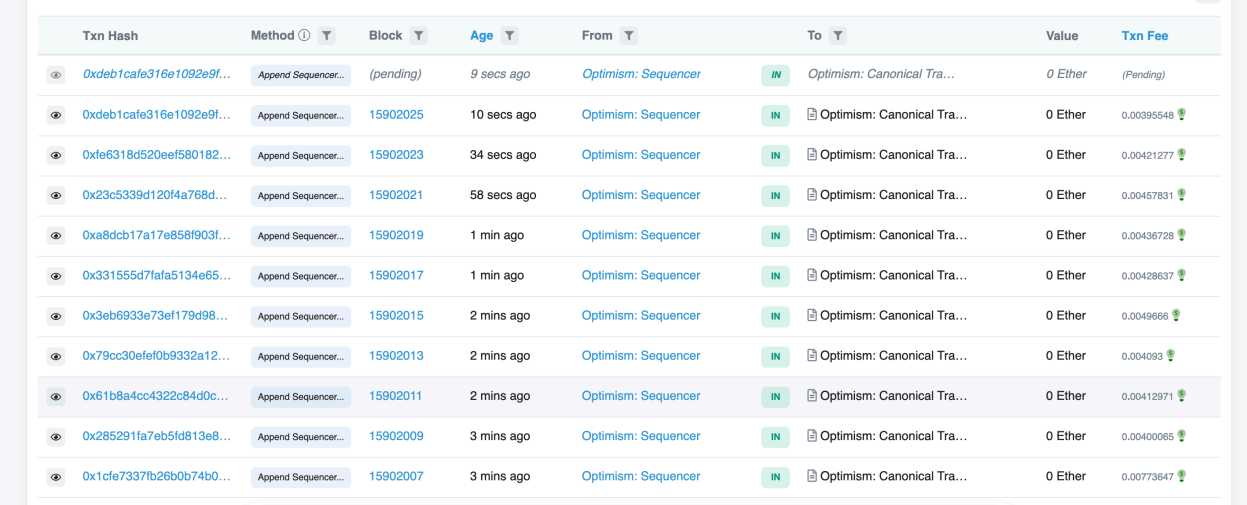
# 1. Write a short write-up describing how you would identify in real-time whether the Optimism bridge is working effectively or not.

So I started off doing research on crypto bridges. Briefly, they allow for the interoperability of blockchains, meaning coins can be “transferred” and their value accurately represented across blockchains to be used in smart contracts, dapps, etc. There’s a variety of mechanisms for this but I found 2 general approaches: lock on the original chain, mint a representation on the other and burn and unlock if you’d like to reverse (lock-mint-burn), or, process transactions off-chain and compress the data into a block on the original chain (Layer 2 solutions).

The bridge we’re looking at specifically is the Optimism bridge for the Optimism blockchain which is a Layer 2 chain built on top of Ethereum, which as the name suggests, functions like the Layer 2 solutions described above. Specifically, Optimism works by storing its blocks in a smart contract on the Ethereum Mainnet (and other blockchains, called the CanonicalTransactioChain (CTC, for short). This CTC Smart Contract is called by another piece of the Optimism Chain, the Sequencer which constructs Layer 2 blocks and manages the process of getting them onto the Ethereum main net, amongst other things.

This covers the case of moving funds from Eth to Optimism. As for the reverse, it's a little different as there is another CTC-like smart contract, the Optimism: State Commitment Chain (SCC, for short) which commits the Optimism blockchain state to Ethereum.

**In order to make sure the Optimism bridge is working correctly, we will focus on monitoring the CTC for consistent, regular transactions from the Sequencer. We will make sure that not too much time has passed between CTC transactions, or else, we could assume that the Bridge is down.**



| Txn Hash | Method | Block | Age | From | | To | Value | Txn Fee |
|---|---|---|---|---|---|---|---|---|
| 0xdeb1cafe316e1092e9f... | Append Sequencer... | (pending) | 9 secs ago | Optimism: Sequencer | IN | Optimism: Canonical Tra... | 0 Ether | (Pending) |
| 0xdeb1cafe316e1092e9f... | Append Sequencer... | 15902025 | 10 secs ago | Optimism: Sequencer | IN | Optimism: Canonical Tra... | 0 Ether | 0.00395548 |
| 0xfe6318d520eef580182... | Append Sequencer... | 15902023 | 34 secs ago | Optimism: Sequencer | IN | Optimism: Canonical Tra... | 0 Ether | 0.00421277 |
| 0x23c5339d120f4a768d... | Append Sequencer... | 15902021 | 58 secs ago | Optimism: Sequencer | IN | Optimism: Canonical Tra... | 0 Ether | 0.00457831 |
| 0xa8dcb17a17e858f903f... | Append Sequencer... | 15902019 | 1 min ago | Optimism: Sequencer | IN | Optimism: Canonical Tra... | 0 Ether | 0.00436728 |
| 0x331555d7fafa5134e65... | Append Sequencer... | 15902017 | 1 min ago | Optimism: Sequencer | IN | Optimism: Canonical Tra... | 0 Ether | 0.00428637 |
| 0x3eb6933e73ef179d98... | Append Sequencer... | 15902015 | 2 mins ago | Optimism: Sequencer | IN | Optimism: Canonical Tra... | 0 Ether | 0.0049666 |
| 0x79cc30efef0b9332a12... | Append Sequencer... | 15902013 | 2 mins ago | Optimism: Sequencer | IN | Optimism: Canonical Tra... | 0 Ether | 0.004093 |
| 0x61b8a4cc4322c84d0c... | Append Sequencer... | 15902011 | 2 mins ago | Optimism: Sequencer | IN | Optimism: Canonical Tra... | 0 Ether | 0.00412971 |
| 0x285291fa7eb5fd813e8... | Append Sequencer... | 15902009 | 3 mins ago | Optimism: Sequencer | IN | Optimism: Canonical Tra... | 0 Ether | 0.00400065 |
| 0x1cfe7337fb26b0b74b0... | Append Sequencer... | 15902007 | 3 mins ago | Optimism: Sequencer | IN | Optimism: Canonical Tra... | 0 Ether | 0.00773647 |

Screenshot of the Optimism CTC from Etherscan. Note the consitent frequency of the Sequencer's transactions to the CTC. Every minute the Sequencer should produce a new transction to the CTC.

**We will do this by using a window of 5 tx's. We will get the timestamp for the first and 5th most recent transactions every 15 seconds and compare that the distance between them is no more than 2 minutes. If this condition fails, we can say that the sequencer of the bridge is non-operational and return False, else True.**

This is, admittedly, a very simplified version of how we could monitor the bridge's liveness. Other approaches that build on top of this could: watch the SCC for a similar frequency of transactions and compare, or we could watch for edge cases related to bridge attacks such as: a large series of failed transactions, very large transaction outflows from the reservoir.

# 2. What all on-chain data would you listen for? Please feel free to read through any docs or research online.

I would watch for transactions being made by the Sequencer to the CTC.

# 3. Write a script (in the language of your choice) to implement your approach and output True if the Optimism bridge is working effectively and vice-versa.

## Link to Code

The monitoring code is written in Golang and made available on my Github.

**Link: https://github.com/obiknows/bridge-monitor**

---

## Resources Used

**Optimism Docs:** https://community.optimism.io/docs/how-optimism-works/#

**(CTC) Optimism: Canonical Transaction Chain:** https://etherscan.io/address/0x5E4e65926BA27467555EB562121fac00D24E9dD2

**(SCC) Optimism: State Commitment Chain:** https://etherscan.io/address/0xBe5dAb4A2e9cd0F27300dB4aB94BeE3A233AEB19

**Paradigm Research's Writeup on Optimism:** https://research.paradigm.xyz/optimism