# Performance and Security Evaluation of Optimism

Bastien Faivre Rachid Guerraoui Gauthier Voron
Ecole Polytechnique Fédérale de Lausanne (EPFL)
*firstname.lastname@epfl.ch*

***Abstract*—Ethereum, with its robust smart contract capabilities, has established itself as a cornerstone of decentralized applications. However, limitations in scalability and high transaction costs have spurred the development of layer 2 (L2) solutions. Optimism, a prominent optimistic rollup solution, aims to address these challenges by processing transactions off-chain and submitting aggregated data to Ethereum.**

**This paper investigates Optimism performance and security characteristics. We benchmark key metrics like throughput and latency against the native Ethereum Proof-of-Stake network. Additionally, we expose a vulnerability by presenting a proof-of-concept censorship attack enabled by the centralized nature of Optimism transaction sequencer.**

**This attack highlights the risks of relying on a single sequencing entity and underscores the urgency of decentralizing this critical component. With billions of dollars currently secured by Optimism, our findings emphasize the significance of ensuring performance and security within the L2 landscape.**

## I. Introduction

Ethereum [7] pioneering role in decentralized applications has solidified its position within the blockchain landscape. However, scalability constraints and associated transaction costs represent ongoing challenges. In response, layer 2 (L2) solutions like rollups have emerged to boost transaction throughput and lower fees. Rollups come in two primary categories:

- **Optimistic Rollups**: These assume transactions are valid by default, employing fraud proofs to challenge potentially fraudulent activity. Optimism [3] and Arbitrum [6] are leading examples of this approach.
- **Zero-Knowledge (ZK) Rollups**: These leverage cryptographic proofs (ZK proofs) to inherently guarantee transaction validity, offering enhanced security but potentially greater complexity. ZKSync [5] and Starknet [2] are such rollups.

Optimism seeks to outperform Ethereum base layer (L1) significantly. Optimism achieves this by bundling transactions off-chain and submitting them to Ethereum in compressed batches. However, Optimism employs a centralized sequencer responsible for transaction ordering to facilitate this efficiency. This design decision introduces a censorship vulnerability, where a compromised sequencer could selectively block transactions. While this vulnerability is acknowledged, the potential for exploitation underscores the need for a practical demonstration and in-depth analysis.

This paper presents the following contributions:

- **Worldwide Performance Benchmarking**: We conduct a worldwide deployment (spanning four continents) of Optimism on an Ethereum Proof-of-Stake network, providing a realistic assessment of Optimism performance in a global context. We measure key performance metrics (throughput and latency) against the barebone Ethereum Proof-of-Stake network to evaluate the potential improvement.
- **Censorship Attack Demonstration**: We design and implement a proof-of-concept censorship attack leveraging a strategically placed Twin proxy [1] to intercept and filter transactions destined for the Optimism sequencer. This attack, deployed globally, highlights the practical implications of sequencer centralization.

The paper structure is as follows: Section 2 presents the technical background of Ethereum and Optimism, outlining their operations and architectural distinctions. Moreover, section 2 presents the benchmarking tool used for the tests. Section 3 details the design and the mechanics of executing the censorship attack through a proxy interception strategy. Section 4 presents the worldwide deployment configuration, the results of our performance benchmarks, and the successful execution of the censorship attack. Finally, section 5 concludes the paper with a summary of findings and insights into the trade-offs between performance and security in L2 rollup design.

## II. Background

### A. Ethereum

Ethereum is a decentralized blockchain platform renowned for its smart contract functionality, enabling the creation of a wide range of decentralized applications (DApps). To secure its network and establish consensus on the order of transactions, Ethereum employs a Proof-of-Stake (PoS) consensus mechanism.

In PoS, rather than relying on energy-intensive mining, individuals become *validators* by staking a significant amount of ETH, the native cryptocurrency of Ethereum. Staking acts as collateral, incentivizing honest behavior within the network. Validators are randomly selected to propose new blocks containing transactions. Other validators then attest to the validity of the proposed block before it is appended to the blockchain. Ethereum PoS system includes incentives and penalties to promote validator integrity and availability. Validators earn rewards for proposing valid blocks and attesting to others, while their staked ETH can be slashed if they engage in malicious behavior or go offline for extended periods.

### B. Optimism

Optimism is a prominent layer 2 (L2) scaling solution built on top of Ethereum. It belongs to the category of optimistic rollups and aims to drastically increase transaction throughput and lower costs while inheriting the fundamental security guarantees of Ethereum.

Optimistic rollups operate by processing transactions off-chain in batches. Instead of directly executing each transaction on the Ethereum base layer (L1), Optimism bundles these transactions and submits a compressed representation (along with a cryptographic proof) to L1. As the name suggests, optimistic rollups initially assume transactions are valid. However, a dispute resolution mechanism based on fraud proofs is in place to challenge potentially fraudulent transactions within a designated challenge period.

The key components of Optimism are:

- **Sequencer**: A specialized node responsible for ordering transactions and submitting periodically transaction batches to Ethereum with the help of a *batcher*. Currently, Optimism relies on a centralized sequencer for efficiency.
- **Fault Proofs**: The mechanism enabling the detection and resolution of fraudulent activity. If a transaction is suspected to be invalid, a fault-proof can be submitted to Ethereum, leading to the transaction being reverted if the challenge is successful.
- **Withdrawal Periods**: When transferring assets back from Optimism (L2) to Ethereum (L1), users must wait for the challenge period to elapse to ensure transaction finality on the L2.
- **Proposer**: a service that publishes transaction results (in the form of L2 state roots) to Ethereum. This aids cross-chain communication, allowing L1 smart contracts to interact with the L2 state changes. Though the proposer role may be revised or eliminated in the future, it is currently an integral component.

Regarding security, Optimism derives from Ethereum. The validity of transactions processed on Optimism can ultimately be verified and enforced on Ethereum through the fraud-proof mechanism. Transaction fees on Optimism are significantly lower than Ethereum L1 fees due to the efficient batching and compression of transactions.

### *C. Diablo*

The rapid proliferation of blockchain platforms with diverse design philosophies presents a challenge for application developers. Choosing the optimal blockchain for a given use case demands a clear understanding of performance and scalability characteristics. Unfortunately, available performance claims of individual blockchain projects are often incomplete, or obtained under isolated and unreproducible test conditions.

Addressing these limitations, the Diablo benchmarking suite [4] aims to standardize and streamline blockchain performance evaluations. There are key components to Diablo design:

- **Realistic DApp Workloads**: Diablo simulates transaction patterns commonly found in real-world decentralized applications. Its initial suite includes representative scenarios from domains such as gaming, web services, exchanges, mobility services, and video sharing.
- **Flexible Deployment**: Diablo facilitates the deployment of blockchain nodes at various scales, enabling performance assessments under differing configurations and network topologies.
- **Comparative Analysis**: Diablo enables cross-chain performance comparisons on an equitable basis by providing a unifying framework.

Extensive benchmark studies using Diablo offer compelling conclusions. Importantly, the Diablo tests highlight that even leading blockchains struggle to fully support the transaction demands of realistic DApps. These detailed analyses pinpoint specific design trade-offs impacting performance, offering valuable guidance for both blockchain developers and those seeking to deploy DApps. Diablo underscores the need for continued optimization and underscores its role as a powerful tool for driving transparent, data-driven advancements in blockchain technology.

## III. Design

To demonstrate the potential for censorship within Optimism centralized sequencer architecture, we implemented a specialized proxy[1] in Golang. This proxy operates as an intermediary between users and the Optimism sequencer, strategically positioned to intercept and manipulate communications.

The proxy intercepts all messages directed towards the sequencer. Requests other than `eth_sendRawTransaction` calls are transparently forwarded to the sequencer without modification, ensuring minimal disruption to regular rollup operations. When the proxy encounters an `eth_sendRawTransaction` request, it performs the following:

1) **Decoding**: The proxy analyzes the request, extracting the raw transaction data.
2) **Signer Identification**: The transaction is parsed to determine the originating address (i.e., the signer public key).
3) **Censorship Logic**: The extracted signer public key is compared against a list of censored addresses. If a match is found, the proxy drops the transaction, effectively preventing it from reaching the sequencer.

This censorship proxy highlights a key vulnerability in centralized sequencer models. It could be deployed either by:

- **Malicious Interceptor**: An external actor positions the proxy strategically to intercept traffic intended for the sequencer.
- **Malicious Sequencer**: The entity operating the sequencer itself incorporates censorship logic directly into the sequencing process.

In both scenarios, the fundamental outcome is the same: the ability to selectively suppress transactions based on arbitrary criteria, such as the transaction originator.

Another potential avenue for censorship would involve directly modifying the sequencer software to incorporate filtering or

[1] https://github.com/BastienFaivre/censorship-proxy

suppression mechanisms. However, this approach would likely require greater technical sophistication and is less relevant for a straightforward proof-of-concept demonstration.

## IV. Evaluation

### A. Settings

To obtain meaningful performance metrics and expose censorship risks in a real-world context, we implemented the Minion-L2 tool[2] to establish a distributed network of Ethereum and Optimism nodes spanning four continents: Europe (*eu-west-2*), North America (*us-east-2*), the Middle East (*me-central-1*), and Asia (*ap-northeast-1*). This deployment aimed to capture the effects of geographic latency on Optimism operations.

Each continental region housed a cluster of five interconnected Ethereum nodes, with two of these nodes running co-located Optimism nodes. All Ethereum nodes formed a fully meshed network to ensure robust connectivity. Similarly, the Optimism nodes established direct peer-to-peer connections. The sole exception was the Optimism sequencer node, located in Europe (*eu-west-2*) and intentionally isolated from other Optimism peers to simulate a censorious environment.

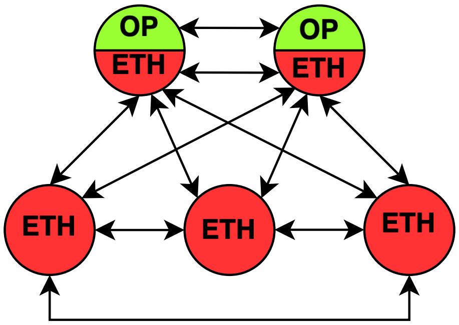

Fig. 1. Node configuration in a continental region

We selected AWS *c6i.2xlarge* instances (*c6in.2xlarge* for the Middle East region due to availability constraints) to provide computational resources for these nodes.

We utilized the Diablo benchmarking tool[3] to simulate realistic transaction workloads. In each region, a dedicated Diablo secondary node orchestrated transaction generation and submission to the local Ethereum and Optimism nodes. A centralized primary Diablo instance, based in Europe (*eu-west-2*), coordinated the global benchmarking process, ensuring synchronization and facilitating results aggregation.

For both Ethereum and Optimism, we established 25 transaction-generating clients per region, totaling 100 clients across the network. This configuration allowed us to investigate performance implications under varying transaction loads. We executed a series of ten benchmark runs, progressively increasing the transaction rate per client. Test cases began at 1 transaction per second (tps) per client and ramped up to a maximum of 10 tps per client. Each test had an active phase of 60 seconds, followed by a 120-second observation period to allow for transaction finalization and comprehensive data collection.

Leveraging the identical network topology, we simulated a censorship attack scenario targeting Optimism. Transaction load was maintained at 2 tps per client. Our custom-developed censorship proxy (detailed in Section 3) was strategically positioned in front of the Optimism sequencer. To demonstrate the censoring capability, we instructed the proxy to selectively block transactions originating from an arbitrary client address beginning at the 30-second mark of each test run. We specifically monitored the targeted client transaction throughput to capture the expected decline in commit rates.

To provide granular insights into performance and censorship effects, the Diablo benchmark tool meticulously tracked the following metrics for every transaction throughout the benchmarks and attack simulation:

- **Submit Time**: The precise timestamp at which a client initiates a transaction.
- **Commit Time**: The timestamp of successful transaction inclusion on-chain (Ethereum L1 or Optimism L2).
- **Abort Time**: The timestamp at which a transaction was deemed invalid or abandoned (if applicable).

From these values, we extract the average throughput between the 5-second (time needed for the system to stabilize) and the 60-second mark using the total number of committed transactions during this time. Moreover, we compute the median latency using values $T_{\text{commit}} - T_{\text{submit}}$ of committed transactions during this range of time.

### B. Results

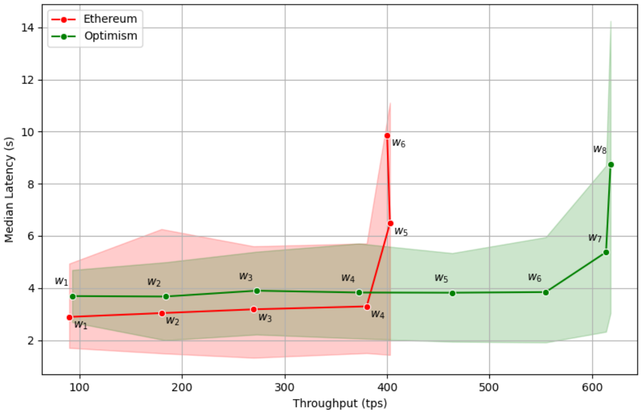

Fig. 2. Benchmark of Ethereum and Optimism with different workloads

*1) Performance:* In Figure 2, we refer to $w_n$ for a workload where all the clients operate at a rate of $n$ transactions per second (tps). In terms of throughput, we observed a better Optimism performance than Ethereum, with an improvement of about $\times 1.5$ $(+50\%)$. Ethereum network starts being saturated at a global workload of 500 tps (100 clients at 5 tps) while Optimism resists until 700 tps. This performance enhancement can likely be attributed to two main key factors:

[2] https://github.com/BastienFaivre/minion-L2

[3] https://github.com/NatoliChris/diablo-benchmark/tree/v2

- **Transaction Batching**: Optimism rollup architecture bundles transactions into compressed batches before submission to Ethereum. This significantly reduces overhead and allows for greater transaction throughput.
- **Optimistic Execution**: By initially assuming transactions are valid, Optimism avoids immediate execution on the Ethereum base layer, contributing to enhanced processing speeds.

However, it is worth noticing that Optimism gets a slightly higher latency on light workloads compared to Ethereum. This likely stems from the additional step of submitting transaction batches to L1 and subsequent L2 block derivation by rollup nodes. This introduces a delay compared to direct transaction processing on Ethereum itself.

Table I reports the detailed results. Note that we omitted some values in Figure 2 for readability purposes.

TABLE I
BENCHMARK OF ETHEREUM AND OPTIMISM WITH DIFFERENT WORKLOADS

| | **Throughput (tps)** | | **Latency (s)** | |
|---|---|---|---|---|
| | **Ethereum** | **Optimism** | **Ethereum** | **Optimism** |
| $w_1$ | 89.98 | 92.97 | 2.89 | 3.69 |
| $w_2$ | 180.00 | 183.77 | 3.04 | 3.67 |
| $w_3$ | 269.97 | 272.70 | 3.18 | 3.90 |
| $w_4$ | 380.25 | 372.58 | 3.29 | 3.83 |
| $w_5$ | 402.93 | 463.68 | 6.48 | 3.82 |
| $w_6$ | 400.20 | 554.90 | 9.86 | 3.84 |
| $w_7$ | 400.20 | 613.77 | 11.34 | 5.37 |
| $w_8$ | 402.00 | 618.15 | 13.82 | 8.75 |
| $w_9$ | 422.43 | 251.33 | 15.80 | 6.80 |
| $w_{10}$ | 419.72 | 132.18 | 17.60 | 6.28 |

*2) Censorship:* In figure 3, we observe that the censorship of the arbitrarily chosen client is successful; its commit rate drops to zero a few seconds after the 30-second mark. Note that the behavior of the other transactions is not affected as expected by the proxy behavior.

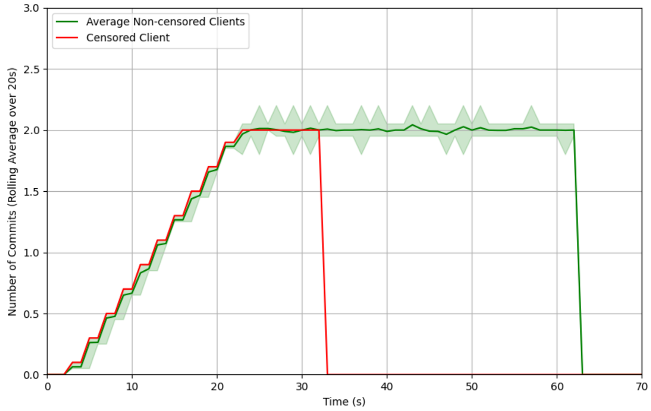

Fig. 3. Censorship attack

## V. Conclusion

Our in-depth study of Optimism has shed light on both its performance potential and critical vulnerabilities. Performance benchmarks demonstrate the scalability gains achievable through Optimism rollup architecture compared to the raw Ethereum network. This enhanced throughput unlocks new possibilities for DApp developers. While our proof-of-concept censorship attack exposed the inherent risks of the original centralized sequencer model, it is important to note that this vulnerability can now be mitigated. The introduction of the OptimismPortal L1 smart contract offers a critical failsafe—users can bypass the sequencer and submit L2 transactions directly, ensuring continued access to the chain even in the face of malicious sequencer behavior.

These findings serve as valuable guides for developers and those considering the geographic distribution of Optimism infrastructure.

We believe our work contributes significantly to the ongoing optimization of L2 blockchain solutions. Despite the mitigation, our research illustrates the necessity of carefully considering potential attack vectors beyond the core consensus layer and underscores the continuous drive for enhanced security and decentralization. We hope this research stimulates broader discussion and further advancements in the development of robust, censorship-resistant, and performance-driven blockchain architectures.

## VI. Acknowledgements

This work is founded by the Ethereum Foundation, Grant ID FY23-0896 — EF ESP - EPFL SCR0750467.

## References

[1] Shehar Bano, Alberto Sonnino, Andrey Chursin, Dmitri Perelman, Zekun Li, Avery Ching, and Dahlia Malkhi. Twins: Bft systems made robust. *arXiv preprint arXiv:2004.10617*, 2020.

[2] The Starknet Community. Starknet. https://www.starknet.io.

[3] The Optimism Foundation. Optimism is Ethereum, scaled. https://github.com/ethereum-optimism/optimism.

[4] Vincent Gramoli, Rachid Guerraoui, Andrei Lebedev, Chris Natoli, and Gauthier Voron. Diablo: A benchmark suite for blockchains. pages 540–556, 2023.

[5] Matter Labs. Zk-Sync. https://github.com/matter-labs/zksync-web-era-docs.

[6] Offchain Labs. Arbitrum Nitro, the latest iteration of the Arbitrum technology. https://github.com/OffchainLabs/nitro/.

[7] Gavin Wood. Ethereum: A secure decentralised generalised transaction ledger, 2015. Yellow paper.