This article has been written by Taiko Cofounder and CEO Daniel Wang.

TL;DR

In this article, we introduce the Based Contestable Rollup (BCR) architecture, examine its advantages and disadvantages, and present our argument for its suitability as the optimal design framework for Ethereum rollups.

Contents

What sets Taiko apart

Proof trustworthiness

Based Contestable Rollup

Multi-proof system

Prover availability

Dynamic configuration adjustments

Cost vs. security tradeoffs

Guardian provers

Our next testnet

What sets Taiko apart

Let’s first start with a look at what sets Taiko apart from its competitors:

1. Permissionless and decentralized

: Taiko is a based rollup, possibly the first of its class. Lacking a centralized sequencer, it relies on Ethereum validators to sequence transactions and blocks, elevating Taiko's permissionless and decentralized ethos.

1. Frictionless developer experience

: Taiko utilizes an Ethereum-equivalent ZK-EVM (type-1) to achieve execution-level compatibility with Ethereum, essentially offering "Ethereum at scale.”

1. Highly configurable and future-compatible

: Designed as a Contestable Rollup, Taiko permits app chains to define their proof systems and to embrace newer, more efficient validity proofs as technology progresses, all without the need to amend Taiko's core protocol.

Permissionless and decentralized

: Taiko is a based rollup, possibly the first of its class. Lacking a centralized sequencer, it relies on Ethereum validators to sequence transactions and blocks, elevating Taiko's permissionless and decentralized ethos.

Frictionless developer experience

: Taiko utilizes an Ethereum-equivalent ZK-EVM (type-1) to achieve execution-level compatibility with Ethereum, essentially

offering "Ethereum at scale."

Highly configurable and future-compatible

: Designed as a Contestable Rollup, Taiko permits app chains to define their proof systems and to embrace newer, more efficient validity proofs as technology progresses, all without the need to amend Taiko's core protocol.

Now let's see why employing a contestation mechanism makes the most sense for Taiko.

Proof trustworthiness

"34,469 lines of code are not going to be bug-free for a long time."

— Vitalik Buterin about the ZK-EVM circuit codes

A pivotal consideration guiding our adoption of the Contestable Rollup and multi-proof structure is a healthy skepticism regarding the infallibility of zero-knowledge proofs (ZKPs). Given that software intricacy frequently increases the probability of bugs — as illustrated by significant oversights like the Heartbleed flaw in OpenSSL (2014) and thecritical Linux Kernel Bug (2003), which went undetected for extended periods — our approach is one of caution.

ZKPs, as a swiftly advancing technology, are prone to errors. Centralized rollups may tolerate certain risks, but Taiko cannot. Aiming for full decentralization and permissionlessness, we envision a future where we relinquish administrative control over the Taiko network. Consequently, we design Taiko with the premise that no validity proof is beyond doubt or wholly secure without years of proven implementation. That's why we need contestation.

Based Contestable Rollup (BCR)

A Based Contestable Rollup is a rollup that features contestation and employsbased sequencing. To illustrate how contestation operates within Taiko's framework, consider this example:

1. Alice proposes a new block.
2. Bob submits a proof for the state transition H1→H2, with H1 being the parent hash and H2 being the hash of the new block. Bob places a 10,000 TKO validity bond. His proof then enters a cooldown period.
3. Bob's proposed state transition and accompanying proof are publicly visible.
4. Cindy, upon noticing an error in Bob's transition, contends it should be H1→H3, not H1→H2. Cindy challenges Bob's proof during the cooldown period by posting her 10,000 TKO contestation bond, yet she does NOT provide an alternative proof or declare the correct transition explicitly.
5. The contested transition is now awaiting a new, higher-tier proof. Both Bob and any other provers have the opportunity to provide this.

Alice proposes a new block.

Bob submits a proof for the state transition H1→H2, with H1 being the parent hash and H2 being the hash of the new block. Bob places a 10,000 TKO validity bond. His proof then enters a cooldown period.

Bob's proposed state transition and accompanying proof are publicly visible.

Cindy, upon noticing an error in Bob's transition, contends it should be H1→H3, not H1→H2. Cindy challenges Bob's proof during the cooldown period by posting her 10,000 TKO contestation bond, yet she does NOT provide an alternative proof or declare the correct transition explicitly.

The contested transition is now awaiting a new, higher-tier proof. Both Bob and any other provers have the opportunity to provide this.

Scenario 1:

- David presents a tier-3 proof for H1→H2, affirming Bob's original claim. David earns a 2,500 TKO reward and becomes the current prover, also posting a 20,000 TKO validity bond.
- Cindy forfeits her entire contestation bond.
- Bob is reimbursed his 10,000 TKO validity bond plus a 2,500 TKO reward.
- David's proof initiates a new cooldown period.

David presents a tier-3 proof for H1→H2, affirming Bob's original claim. David earns a 2,500 TKO reward and becomes the current prover, also posting a 20,000 TKO validity bond.

Cindy forfeits her entire contestation bond.

Bob is reimbursed his 10,000 TKO validity bond plus a 2,500 TKO reward.

David's proof initiates a new cooldown period.

Scenario 2:

- David provides a tier-3 proof for a transition from H1→H4, which indicates Bob’s transition was incorrect. David receives a 2,500 TKO reward and secures his position with a 20,000 TKO validity bond.
- Cindy retrieves her 10,000 TKO contestation bond and an additional 2,500 TKO reward.
- Bob’s validity bond is seized.
- A new cooldown period commences for the fresh proof.

David provides a tier-3 proof for a transition from H1→H4, which indicates Bob’s transition was incorrect. David receives a 2,500 TKO reward and secures his position with a 20,000 TKO validity bond.

Cindy retrieves her 10,000 TKO contestation bond and an additional 2,500 TKO reward.

Bob’s validity bond is seized.

A new cooldown period commences for the fresh proof.

Each proof in Taiko, except for the highest tier, requires a validity bond

paid in Taiko tokens by the original prover. This proof enters a cooldown window, during which it can be contested by others who do not need to provide any fraud/validity proofs but must also pay a contestation bond

in Taiko tokens. If contested, a higher-tier proof is required to resolve the dispute before this block can be verified.

- If the contester wins

: The contester receives the contestation bond back and also gets 1/4 of the original prover's validity bond. The new prover receives 1/4 of the original prover's validity bond as a proving fee, and the remaining 1/2 is forfeited.

- If the original prover wins

: The original prover reclaims the validity bond and gets 1/4 of the contestation bond as a reward. The new prover (who may be the original prover) earns 1/4 of the contestation bond, and the remaining 1/2 is forfeited.

If the contester wins

: The contester receives the contestation bond back and also gets 1/4 of the original prover's validity bond. The new prover receives 1/4 of the original prover's validity bond as a proving fee, and the remaining 1/2 is forfeited.

If the original prover wins

: The original prover reclaims the validity bond and gets 1/4 of the contestation bond as a reward. The new prover (who may be the original prover) earns 1/4 of the contestation bond, and the remaining 1/2 is forfeited.

The new prover must also pay a validity bond according to the new tier's rules unless they're providing the highest-tier proof, in which case the state transition is considered final, and no further contestation is allowed.

It’s worth noting that the contestation/proving game assumes a correct parent block hash. If the parent block hash is incorrect, the winning transition won't be used for block verification, and the prover will forfeit their validity bond entirely.

Repeated rounds of contestation and proving extend the time required to verify a block. Each new round introduces its own proving

and cooldown windows

. However, since contestation involves financial stakes with definitive losses for the losers, frequent contestations are unlikely. Additionally, the validity and contestation bonds for higher-tier proofs are significantly larger than those for lower tiers. As the game progresses through a few rounds, the associated costs can escalate dramatically, further discouraging frivolous contestation.

A potential drawback of this design is that it does economically disincentivize the submission of invalid yet verifiable proofs, making bug detection more challenging. However, one could question the wisdom of using the whole blockchain as a bug-finding stake. Alternative mechanisms could be employed to encourage reporting of such bugs. For instance, offering substantial rewards to those who identify invalid but verifiable proofs could be a better approach, rather than jeopardizing

user assets.

Multi-proof system

The multi-proof feature is integral to Taiko's BCR architecture, allowing each tier to use its proving system. While ranking these systems by trustworthiness may seem subjective, it is theoretically possible to construct proofs that are more reliable than others.

For instance, we could create a composite prover C that combines proofs from provers A and B. A state transition is considered proven by C only if both A and B prove it as such. Although this increases costs, it also enhances security. One limitation of this approach is that a composite proof of type C relies on the successful generation of sub-proofs from A and B. If either A or B contains bugs, generating a reliable type C proof becomes problematic.

In practice, multi-proof tiers are often subjectively configured. For example, it's reasonable to assume that an SGX prover is more trustworthy than an optimistic prover, which lacks actual proof. A ZK prover is arguably more secure than an SGX prover, and a hybrid SGX+ZK prover would rank even higher

.

If a ZK proving system eventually proves to be universally secure, the Contestable Rollup can be configured to use this single tier, effectively transitioning to a conventional, non-contestable ZK-Rollup.

In the context of Taiko's Contestable Rollup design, the multi-proof serves dual functions: Vertically, it enables a tier-based architecture; horizontally, it allows for the composition of multiple sub-provers to a composite prover that has a lower likelihood of false positives.

Prover availability

A potential vulnerability of the contestation design is the lack of active higher-tier provers, especially since contestations are infrequent. To maintain a pool of available high-tier provers, Taiko introduces a mechanism that randomly assigns a minimum required tier for each new block

. The likelihood of a block being assigned a higher tier is inversely proportional to the tier level; for instance, only 5% of blocks might require SGX+ZKP, while 20% demand ZKP, leaving the majority to require just SGX. This ensures that ZK provers always have work to keep them engaged and profitable.

An attack vector for the Contestable Rollup is a capital-intensive proof submission aimed at draining higher-tier proving resources. While such an attack may slow down block finalization and proposal rates, it's unlikely to compromise the chain's overall security. This is because community nodes can collectively contest invalid proofs by placing contestation bonds. Importantly, contesters don't need to provide any proofs themselves, but attackers must offer on-chain verifiable proofs for each block. Generating faulty yet verifiable proofs is likely more challenging and less cost-effective than generating correct ones.

If higher-tier proofs suddenly become necessary, the financial incentives should attract new provers. Specifically, receiving 1/4 of the validity and contestation bonds would likely represent a more lucrative opportunity than regular proving fees. For example, ZK provers working across multiple platforms will likely switch over to tackle blocks in attestation for a higher profit, given the larger financial stakes involved.

Dynamic configuration adjustments

One of the advantages of Taiko's contestable design is its adaptability. As the cost of higher-tier proofs decreases, the system can dynamically adjust the proportion of blocks requiring ZK proofs without impacting existing blocks or necessitating a protocol upgrade.

For example, an extreme scenario could start with 100% of blocks being optimistic, then switch or gradually transition to 100% requiring ZK proofs for minimizing time to on-chain finality. This enables rollups, L2s or L3s, built on top of Taiko’s protocol stack to evolve from something like an Optimistic Rollup to a ZK-Rollup, offering significant flexibility in optimizing both security and economic incentives.

Cost vs. security tradeoffs

In the landscape of rollups, concerns about game theory's implications are understandable. Yet, the introduction of a contestation layer does not necessarily imply that a rollup is less secure. A traditional rollup with a set ZK prover is no less secure than a Contestable Rollup using that same prover as the base tier.

ZK-Rollups provide superior security, but in the short term, the associated costs may not be sustainable for applications with millions of daily users. As Vitalik observed, "It's easier to justify paying $0.10 if you were paying $1 before than if you were paying $0 before."

In the next few years, most ZK-Rollups will still be financially impractical for high-transaction-volume app chains. Faced with this, many app chains are likely to prioritize cost-efficiency over enhanced security. Even if ZK proving costs are expected to decrease, it may not be enough to entice app chains to fully transition from more economical but centralized solutions.

Contestable Rollups provide a middle ground, enabling app chains to start with more economical configurations while retaining the flexibility to scale up security measures incrementally, without necessitating significant alterations to their existing infrastructure.

Guardian provers

Guardian provers are multisig signers who collectively serve as the higher tier in the proof hierarchy during the first couple of years after launch. These guardian provers serve a specific purpose: To act as a safety net for addressing bugs in the proving system during its early stages. Importantly, these guardian provers live outside the core protocol and can be removed from the tiered proving configuration. They cannot influence the sequencing of transactions or blocks. As the system matures and other provers prove to be reliable, the role of these guardian provers can be phased out.

While alternative mechanisms like DAO voting could manage faulty provers, such an approach would require significantly longer cooldown windows to complete the governance process. This would be unsuitable for promptly addressing critical bugs, unlike adding or removing features, which can afford a more lengthy, decentralized governance process.

Guardian provers are crucial during the initial phases of a rollup's deployment, especially if the goal is decentralization. Unlike in centralized rollups, where the chain can be paused or altered by its administrators, guardian provers offer a security layer that can address errors or vulnerabilities without undermining the network's decentralized nature.

Our next testnet

We’re currently fine-tuning our new design and the rollup contract codebase. Upon completion, this will be deployed as Taiko’s Alpha-6 testnet, featuring a four-tier proof system. The percentages in the below diagram represent the probability that a block will require a given tier as its minimum proof level.

In the upcoming A6 testnet, we’ll employ a cooldown period of 24 hours for all tiers.

Finally, our primary aim is for Contestable Rollups to combine the strengths of Optimistic Rollups and ZK-Rollups. If you're keen to see how this innovative design unfolds in practice, we invite you to join our community for further updates!

Join us

Explore open positions on our job board.

Follow us

Get the latest from Taiko:

- Website: https://taiko.xyz.
- Discord: https://discord.gg/taikoxyz.
- GitHub: https://github.com/taikoxyz.
- Twitter: https://twitter.com/taikoxyz.
- Community forum: https://community.taiko.xyz.
- Youtube: https://www.youtube.com/@taikoxyz.

Website: https://taiko.xyz.

Discord: https://discord.gg/taikoxyz.

GitHub: https://github.com/taikoxyz.

Twitter: https://twitter.com/taikoxyz.

Community forum: https://community.taiko.xyz.

Youtube: https://www.youtube.com/@taikoxyz.

Contribute

Contribute to Taiko on GitHub and earn a GitPOAP! You will also be featured as a contributor on our README. Get started with the contributing manual.