Hello Optimism Collective!

Excited to share plans on a project our team has been working on over the past few weeks.

Our goal is simple - we'd like to add private, bribery-resistant, on-chain voting to Optimism's RetroPGF rounds.

Who are we?

We're the core team working on MACI, an on-chain voting platform which protects privacy and minimizes the risk of collusion and bribery. We're an engineering team within PSE, which is supported by the Ethereum Foundation

You can learn more about MACI in ourdocumentation.

What are we doing?

We're building a proof-of-concept RPGF implementation that supports private on-chain voting for badgeholders of the Citizens House.

- Thanks to folks at Gitcoin & their work onEasyRetroPGF, we have a purpose-built web app that we're forking for our frontend
- For the "backend", we're essentially swapping out the SQL database for the MACI smart contracts
- For badgeholder verification, we've built an EAS gatekeeper contract which ensures only OP badgeholders are eligible to register to vote
- In line with PSE's mission and inspired by the efforts of EasyRetroPGF, we also open-source all our code in case it's beneficial for others

Here's the source code:

GitHub

GitHub - privacy-scaling-explorations/maci-rpgf: RPGF with MACI

RPGF with MACI. Contribute to privacy-scaling-explorations/maci-rpgf development by creating an account on GitHub.

Here's where were tracking our progress:

GitHub

MACI team sprint board • privacy-scaling-explorations

MACI team sprint board

Why are we doing this?

In many ways, Optimism's RPGF rounds are already a massive success. It seems the process worked in round 3 but - like many of you - we see several areas that can be improved.

For one, the RPGF process currently depends heavily on the OP foundation to act honestly and competently. In round 3, the community had to trust that:

- Only badgeholders would be able to submit ballots
- No badgeholder ballots would be censored or manipulated or double-counted
- The final tally of ballots would be calculated correctly

If we think about what security properties are critical to a voting, Vitalik outlined this in his blog post(Blockchain voting is overrated among uninformed people but underrated among informed people) as well as anyone could:

Any voting system requires a few crucial security properties in order to be trusted by users:

- Correct execution

: the results (tally of votes) must be correct, and the results must be guaranteed by a transparent process (so that everyone is convinced that the results are correct)

- Censorship resistance

: anyone eligible to vote should be able to vote, and it should not be possible to interfere with anyone's attempt to vote or to prevent anyone's vote from being counted

- Privacy:

no one should be able to tell which candidate anyone voted for, or if they even voted at all

- Coercion resistance:

you should not be able to prove to someone else how you voted, even if you want to

Unfortunately RPGF currently fails in all 4 dimensions. As RPGF continues to grow, we think the need to enforce process integrity will be of upmost importance.

Talking with the folks from the OP foundation about, they've mentioned some key requirements for future versions of their RPGF stack:

- Provably correct execution
- They'd want to demonstrate the legitimacy to community in terms of how ballots are submitted and calculated
- Privacy protection
- They want to ensure not just voter privacy, but also vote privacy
- Why? While voter privacy alone would offer some valuable protection, if the votes are public, there's the chance that data sleuths would be able to deduce the identity of some badgeholders. e.g. say a badgeholder publicly declares a few projects where they have conflicts of interest. If they are the only badgeholder who does NOT vote for those specific projects, anyone would be able view their public ballot and tie it to that specific badgeholder, thus circumventing this supposed privacy protection
- Why? While voter privacy alone would offer some valuable protection, if the votes are public, there's the chance that data sleuths would be able to deduce the identity of some badgeholders. e.g. say a badgeholder publicly declares a few projects where they have conflicts of interest. If they are the only badgeholder who does NOT vote for those specific projects, anyone would be able view their public ballot and tie it to that specific badgeholder, thus circumventing this supposed privacy protection
- Collusion resistance
- given the money at stake with RPGF, bribery resistance is a key component for any RPGF implementation

What solutions exist?

We reviewed the options in this private voting report:State of Private Voting

It's great to see an array of private on-chain voting solutions emerging! As you can see from the report, each has a unique set of features and trade-offs with their implementation.

[

Image 2024-02-28 at 15.00.57

1470×734 205 KB

](https://global.discourse-cdn.com/business7/uploads/bc41dd/original/2X/4/4ccf0456617e88907585beae15e19bbef5ac875d.jpeg)

Why did we decide on this implementation?

A number of projects solve some of the above requirements, but again - given the money at stake with RPGF, we feel that collusion resistance (”Briber Protection”, in that report), is an essential component for any RPGF implementation. Currently MACI is the only project with all 3 of these features:

[

Image 2024-02-26 at 19.13.39

1326×640 179 KB

](https://global.discourse-cdn.com/business7/uploads/bc41dd/original/2X/b/b8caf735f0f4842ffaafa6270dd3b756a6ce5977.jpeg)

An RPGF MACI integration has the potential to provide several important security guarantees to the RPGF voting process:

1. Correct execution:

With MACI, user registration, voting data and poll logic is stored on-chain. While tallying computation is handled off-chain, ZK-proofs guarantee the correct execution of this logic. In this way, we know the result (tally of votes) will be correct, and the results are guaranteed by a transparent process (anyone can verify that the result is correct) .

1. Censorship resistance

: With voter verification (via an EAS attestation gatekeeper) and vote submission happening on-chain, there's no way for anyone (including the OP foundation) to censor any badgeholder votes.

1. Privacy:

With MACI's receipt-free voting scheme, we're able to ensure that results are transparent, but it is impossible for outsiders to verify how any specific user voted (since on-chain votes are encrypted). Vote tallying takes place off-chain but ZKPs are submitted and verified on-chain, which guarantees votes are counted correctly without revealing the individual votes.

1. Coercion resistance:

With MACI's private, receipt-free votes, this makes cheating (like bribery) much harder. User's cannot prove which option they voted for, and therefore bribers cannot reliably trust that a user voted for their preferred option. This prevents any bribers from simply reading the transaction data to see which option a user voted for.

Input/Feedback?

We'd love to hear feedback on this plan! The implementation is in-progress but we're happy to make potential adjustments based on community input.

If there are any badgeholders interested in being user-testers for us, please let us know and we'll get in touch to schedule an user interview once we have a working demo up and running on a testnet! Feel free to comment here, hop into the PSE Discord (# -maci

channel), hit us up on Twitter/X, or reach out to me directly (sam at ethereum dot org).

Thanks!

Sam