Mode GovernanceToken

Security Review

Review by:
Alex the Entreprenerd, Security Researcher

April 29, 2024

Contents

1 Introduction

1.1 Disclaimer

A security review a detailed evaluation of the security posture of the code at a particular moment based on the information available at the time of the review. While the review endeavors to identify and disclose all potential security issues, it cannot guarantee that every vulnerability will be detected or that the code will be entirely secure against all possible attacks. The assessment is conducted based on the specific commit and version of the code provided. Any subsequent modifications to the code may introduce new vulnerabilities that were absent during the initial review. Therefore, any changes made to the code require a new security review to ensure that the code remains secure. Please be advised that a security review is not a replacement for continuous security measures such as penetration testing, vulnerability scanning, and regular code reviews.

1.2 Risk assessment

Severity	Description
Critical	*Must* fix as soon as possible (if already deployed).
High	Leads to a loss of a significant portion (>10%) of assets in the protocol, or significant harm to a majority of users.
Medium	Global losses <10% or losses to only a subset of users, but still unacceptable.
Low	Losses will be annoying but bearable. Applies to things like griefing attacks that can be easily repaired or even gas inefficiencies.
Gas Optimization	Suggestions around gas saving practices.
Informational	Suggestions around best practices or readability.

1.2.1 Severity Classification

The severity of security issues found during the security review is categorized based on the above table. Critical findings have a high likelihood of being exploited and must be addressed immediately. High findings are almost certain to occur, easy to perform, or not easy but highly incentivized thus must be fixed as soon as possible.

Medium findings are conditionally possible or incentivized but are still relatively likely to occur and should be addressed. Low findings a rare combination of circumstances to exploit, or offer little to no incentive to exploit but are recommended to be addressed.

Lastly, some findings might represent objective improvements that should be addressed but do not impact the project's overall security (Gas and Informational findings).

2 Security Review Summary

Mode is the Ethereum L2 that rewards users for growing the network via new economic mechanisms. Built on the OP Stack L2, designed for growth that incentivises and directly rewards developers, users and protocols to grow Mode and the Superchain ecosystem.

From Apr 30th to May 1st the security researchers conducted a review of GovernanceToken.sol as seen on address 0xDfc7C877a950e49D2610114102175A06C2e3167a in the Mode network. a total of **4** issues in the following risk categories:

- Critical Risk: 0
- High Risk: 0
- Medium Risk: 0
- Low Risk: 0
- Gas Optimizations: 0
- Informational: 4

3 Findings

3.1 Informational

3.1.1 Max mint will be slightly different than OP token

Severity: Informational

Context: GovernanceToken.sol

Description: Due to the upgrade to OpenZeppelin 5, the maximum total supply for the Mode Token can be a `uint224`. Differential fuzzing revealed this, here's an example repro:

```
function test_op_governanceToken_mint_() public {
    op_governanceToken_mint(
        0xffffffffffffffffffffffffffffffffffffffff,
        171269723124715185726994316333939416365929594880000000000000000000
    );
}
```

This should not be an issue in practice.

3.1.2 Compiler is locked at `0.8.20`

Severity: Informational

Context: GovernanceToken.sol

Description: The compilation process uses Solidity `0.8.20` version, but files have a floating pragma. You can remove the `^` to use a fixed compiler version.

Recommendation: Remove the `^` to use a fixed compiler version.

3.1.3 No max total supply is enforced in the contract

Severity: Informational

Context: GovernanceToken.sol

Description: The contract doesn't cap total supply. The hard cap due to implementation is `uint224`.

3.1.4 Admin can mint without restrictions

Severity: Informational

Context: GovernanceToken.sol

Description: Restrictions can be enforced by a timelock, or a mint schedule, which can be added externally.

Mitigated by using a 3/5 Gnosis Safe. See address 0xaa9703bea2aae3e6db568d20fb16caad3096fdf8.

Recommendation: Add a timelock as this currently offers no guarantees to holders.

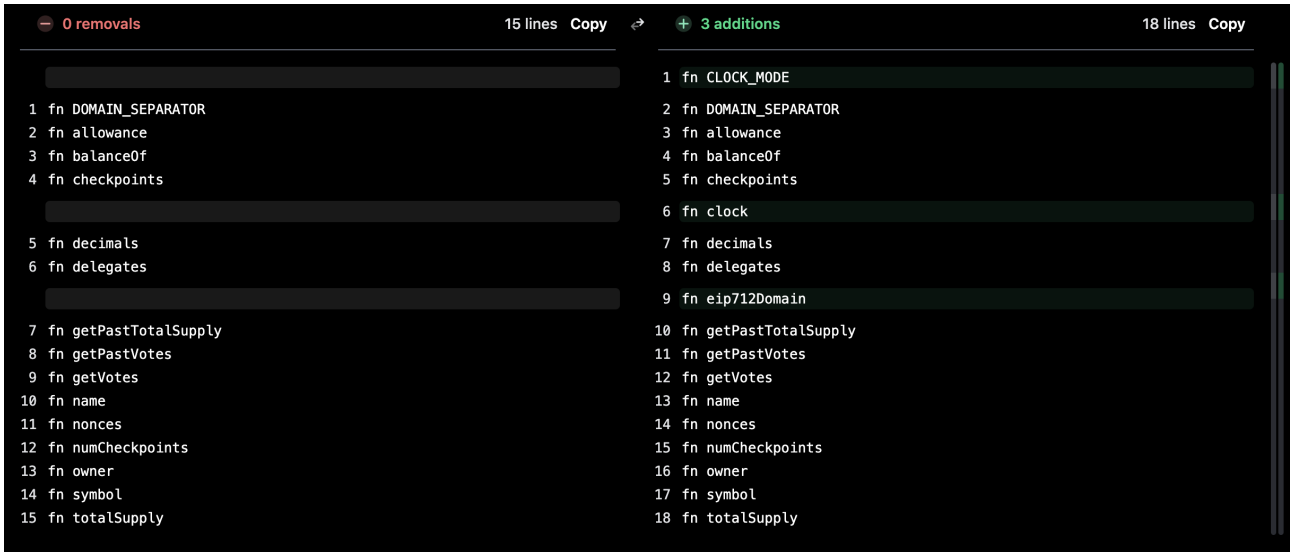
4 Appendix

4.1 Analysis

The Mode token is a fork of the OP token that uses Solidity 0.8.20 and Open Zeppelin V5. No substantial change has been brought forward by the Mode Token

4.1.1 Old ABI View Functions

A few extra View functions are introduced by the upgrade to OZ V5:



They do not pose a security risk.

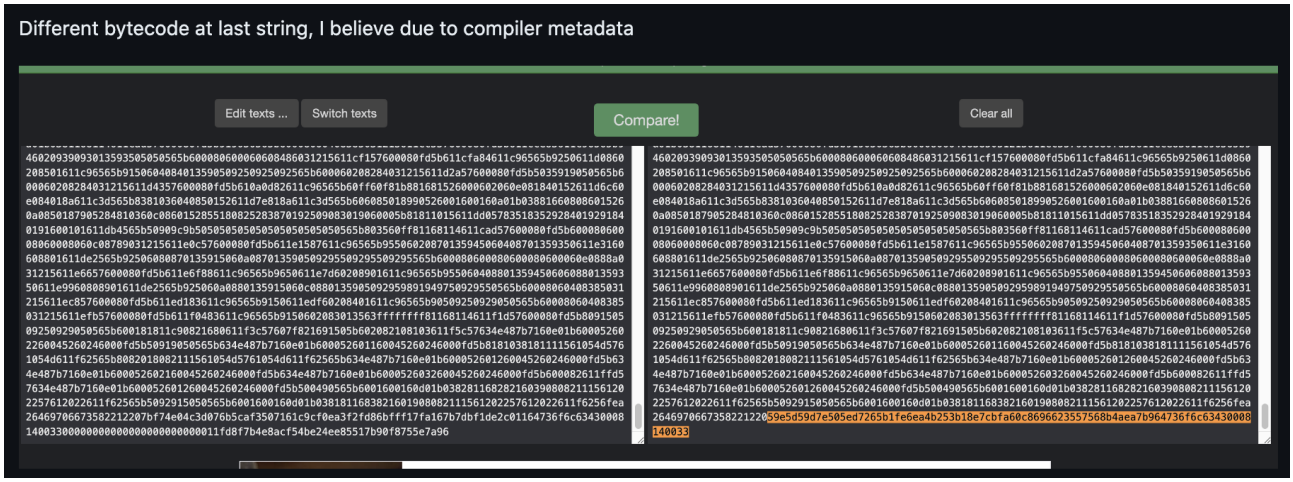
4.2 Differential Fuzzing against OP Token

4.2.1 Against OP Code

To demonstrate the compatibility of the two tokens, a differential run against the deployed OP bytecode was performed. The logs from the run are available here.

4.2.2 Of Compiled Bytecode against Mode Deployed Bytecode

The bytecode I was able to compile locally matches the deployed bytecode minus a final string, which I believe is the compiler metadata:



I ran differential fuzzing of the two bytecodes in a similar way to the test against the OP contract. The logs from the run are available here.