Quantstamp

M^0

Executive Summary

This audit report was prepared by Quantstamp, the leader in blockchain security.

Type	Two-Token Governance & Collateralized Token
Timeline	2024-01-08 through 2024-01-29
Language	Solidity
Methods	Architecture Review, Unit Testing, Functional Testing, Computer-Aided Verification, Manual Review
Specification	None
Source Code	• MZero-Labs/spog #a812790 • https://github.com/MZero-Labs/protocol #3499f50 • https://github.com/MZero-Labs/common #4a37119 • MZero-Labs/ttg #930f2db • MZero-Labs/protocol #44784c6 • MZero-Labs/common #e809402
Auditors	• Jennifer Wu Auditing Engineer • Rabib Islam Auditing Engineer • Roman Rohleder Senior Auditing Engineer

Documentation quality	High
Test quality	High
Total Findings	21 Fixed: 8 Acknowledged: 10 Mitigated: 3
High severity findings	0
Medium severity findings	1 Fixed: 1
Low severity findings	9 Fixed: 3 Acknowledged: 3 Mitigated: 3
Undetermined severity findings	0
Informational findings	11 Fixed: 4 Acknowledged: 7

Summary of Findings

Initial Audit

The present report is concerning an audit conducted on three repositories to do with the M^Zero project. This project has two major functional components. The primary component, comprising the main goal of the project, is to enable holders of off-chain collateral to mint a token (M) on-chain (the Protocol). The secondary component is a governance system that involves two different tokens and the ability to propose governance actions and vote on those proposals (Two-Token Governance aka TTG). Each of these components is contained within its own respective repository. The third repository consists of contracts that are common to both functions (hence Common). All three repositories work in tandem to form the M^0 system.

Overall, the quality of the code is good: it is well-structured and it effectively carries out the stated purposes of the protocol while attempting to minimize gas costs. Moreover, the test suite is extensive and the documentation is very descriptive.

The major issues found during this audit were mainly to do with potential for integer overflow and integer truncation during casting. These issues typically are at risk of being exploited only when certain storage quantities reach very high values. As such, although these issues may be quite damaging if triggered, we expect the likelihood of their exploitation to be relatively low.

Also of note is that the protocol's proper operation relies on a significant off-chain component. Validators have very strong permissions on-chain, including the power to cancel mint operations and freeze minters. Validators are, however, chosen through governance. The reliability of the protocol is therefore crucially dependent on an interplay between economic incentives and communal trust, in stark contrast to those protocols that attempt to rely strictly on smart contracts for operation.

Update

The client has addressed all the issues. We have also adjusted a number of the findings' severities.

ID	DESCRIPTION	SEVERITY	STATUS
MZ-1	Returning Excessively High Unrealized Inflation in Exceptional Scenario	Medium	Fixed
MZ-2	Risk of Overflow when Minting M Tokens	Low	Fixed
MZ-3	Truncation Risk in Arbitrary Integer Casting	Low	Fixed
MZ-4	Privileged Roles and Ownership	Low	Acknowledged
MZ-5	Overflow Risks in Unchecked Arithmetic Operations	Low	Mitigated
MZ-6	Precision Loss in Token Distribution Affects Small Holders	Low	Mitigated
MZ-7	Power Token Supply Can Be Bypassed Through Re-Entrancy	Low	Fixed
MZ-8	Ambiguous Error Messaging in `MinterGateway.proposeMint()`	Informational	Acknowledged
MZ-9	Signatures Missing Expiry	Low	Acknowledged
MZ-10	Functions for Getting Vote Token Delegatees and Vote Power Amounts May Revert if Gas Too High	Low	Acknowledged
MZ-11	Missing Input Validation	Low	Mitigated
MZ-12	Rounding Mismatch in M Token Minting and Debt Calculation	Informational	Acknowledged
MZ-13	Risks in Validator-Based Collateral Attestation with Off-Chain Dependencies	Informational	Acknowledged
MZ-14	Risk of Standard Governance Griefing Through Excessive Low-Cost Proposals	Informational	Acknowledged
MZ-15	Limitation of Padé Approximant in Approximating Exponential Functions for Financial Calculations	Informational	Acknowledged
MZ-16	Loss of Precision Due to Division before Multiplication	Informational	Fixed
MZ-17	Clone-and-Own	Informational	Acknowledged
MZ-18	Code Documentation	Informational	Fixed
MZ-19	Undocumented Magic Constants	Informational	Fixed
MZ-20	Outstanding Todo Comments	Informational	Fixed
MZ-21	Adherence to Specification	Informational	Acknowledged

Assessment Breakdown

Quantstamp's objective was to evaluate the repository for security-related issues, code quality, and adherence to specification and best practices.

> **Disclaimer**
>
> Only features that are contained within the repositories at the commit hashes specified on the front page of the report are within the scope of the audit and fix review. All features added in future revisions of the code are excluded from consideration in this report.

Possible issues we looked for included (but are not limited to):

- Transaction-ordering dependence
- Timestamp dependence
- Mishandled exceptions and call stack limits
- Unsafe external calls
- Integer overflow / underflow
- Number rounding errors
- Reentrancy and cross-function vulnerabilities
- Denial of service / logical oversights
- Access control
- Centralization of power
- Business logic contradicting the specification
- Code clones, functionality duplication
- Gas usage
- Arbitrary token minting

Methodology

1. Code review that includes the following
    1. Review of the specifications, sources, and instructions provided to Quantstamp to make sure we understand the size, scope, and functionality of the smart contract.
    2. Manual review of code, which is the process of reading source code line-by-line in an attempt to identify potential vulnerabilities.
    3. Comparison to specification, which is the process of checking whether the code does what the specifications, sources, and instructions provided to Quantstamp describe.
2. Testing and automated analysis that includes the following:
    1. Test coverage analysis, which is the process of determining whether the test cases are actually covering the code and how much code is exercised when we run those test cases.
    2. Symbolic execution, which is analyzing a program to determine what inputs cause each part of a program to execute.
3. Best practices review, which is a review of the smart contracts to improve efficiency, effectiveness, clarity, maintainability, security, and control based on the established industry and academic practices, recommendations, and research.
4. Specific, itemized, and actionable recommendations to help you take steps to secure your smart contracts.

Scope

The `src` directory for each repository is within scope. The remainder of each repository is out of scope.

Note that the `spog` repository has been renamed to `ttg`.

Files Included

- `MZero-Labs/spog` : `src/*`
- `MZero-Labs/protocol` : `src/*`
- `MZero-Labs/common` : `src/*`

Findings

MZ-1

Returning Excessively High Unrealized Inflation in Exceptional Scenario

Medium | Fixed

> **Update**
>
> Marked as "Fixed" by the client.
> Addressed in: `0c86903456266e3814d5086c63b680063d03bdb6`.

File(s) affected: `EpochBasedInflationaryVoteToken.sol`

Description: At `EpochBasedInflationaryVoteToken.sol#L279`, the function `_getUnrealizedInflation()` returns `type(uint240).max` as the appropriate inflation amount. However, value is being returned on the basis of the inflated balance being above `type(uint240).max`, not that the actual inflation is above `type(uint240).max`. As a result, the returned inflation value may in fact be significantly larger than the actual inflation value determined as a function of time and balance.

Furthermore, due to the fact that unrealized inflation amounts are used to add to the existing balance using `EpochBasedVoteToken._addBalance()`, wherein the adding operation is `_addUnchecked()`, this will lead to integer overflow when `inflatedBalance_` is sufficiently high. Because this arithmetic is occurring in the context of a user's token balance, this may lead to an unintended burn that negatively impacts the user as well as affecting the wider economy in an unexpected way.

Recommendation: Instead of returning `type(uint240).max` in this scenario, consider returning the maximum `inflation_` value that would lead to `inflatedBalance_` being equal to `type(uint240).max`.

MZ-2 Risk of Overflow when Minting M Tokens

Low Fixed

Update
Marked as "Fixed" by the client.
Addressed in: `879a975e67f7be84733ae47e33fdadecd4756173` and `ef0dc6d2d7b4b4277e8886c17134cbb02a20cef3`.

File(s) affected: `MinterGateway.sol`, `MToken.sol`

Description: The `MinterGateway.mintM()` function calculates the `principalAmount` for minting M tokens based on the current index of `MinterGateway` and the amount to be minted. This amount is tracked by `principalOfTotalActiveOwedM`, which checks if the new total principal amount exceeds `type(uint112).max`. If within limits, minting is allowed. However, a potential risk of overflow exists within the `MToken.mint()` function. It performs unchecked arithmetic operations to add the minted amount to `principalOfTotalEarningSupply` or `totalNonEarningSupply`, based on the recipient's earning status. This happens through `_addEarningAmount()` and `_addNonEarningAmount()` functions, which are unchecked arithmetic and can overflow. Although there is a check in the function `MToken._mint()` to validate that `principalOfTotalEarningSupply + _getPrincipalAmountRoundedDown(totalNonEarningSupply) < type(uint112).max`, this may not prevent overflow if `principalOfTotalEarningSupply` or `totalNonEarningSupply` has already exceeded its limit due to unchecked addition.

When minting through the `MinterGateway`, the following concerns are:

1. The index used by `MinterGateway` and `MToken` may differ, leading to a potential overflow when computing the principal amount. The post-addition check within `MToken._mint()` might fail to catch this overflow.
2. After minting to the recipient, the `MinterGateway.updateIndex()` function additionally mints tokens to the `ttgVault` without performing an M token supply check. This absence of a pre-minting check might lead to an oversight of potential overflow within `MToken._mint()`. While the supply minted to `ttgVault` is classified as non-earning (`totalNonEarningSupply` of type `uint240`), the subsequent distribution of these tokens to Zero token holders poses a risk. Specifically, when Zero token holders transition to earning status via `startEarning()`, their principal balance is added to `principalOfTotalEarningSupply`. This transition can result in an overflow of `principalOfTotalEarningSupply`.

Recommendation: Remove unchecked arithmetic in `_addEarningAmount()` and `_addNonEarningAmount()` within the `MToken.mint()` function.

MZ-3 Truncation Risk in Arbitrary Integer Casting

Low Fixed

Update
Marked as "Fixed" by the client.
Addressed in: `fdb93ed64fd40294cac9935574d6a038bfacc289`.

File(s) affected: `EpochBasedInflationaryVoteToken.sol`, `ContinuousIndexingMath.sol`

Description: The protocol employs the casting of variables between integer types of different sizes (e.g., casting from `uint256` to `uint240`). While this practice is often used to save on storage costs or align with specific interface requirements, it inherently carries the risk of truncation if the original value exceeds the maximum representable value of the target type.

The following instance carries the risk of truncation from integer casting:

1. In the function `ContinuousIndexingMath.convertToBasisPoints()`, the converted basis point can exceed the maximum value of `uint32`.

Although this instance is not going to lead to an issue in the current codebase, we note this because of the possibility that the code is used in a modified or different codebase; as used presently, the input being `uint64` is unnecessary, because the argument at `StableRateEarnerModel.sol#L116` is a `uint32` cast to `uint64`, presumably to accord with the function.

Recommendation: Wherever type casting is performed, introduce checks to ensure that the value being cast does not exceed the capacity of the target type. Consider using or implementing a safe casting library (e.g., OpenZeppelin's) that handles these checks automatically.

MZ-4 Privileged Roles and Ownership

Low Acknowledged

Update

Marked as "Acknowledged" by the client.
The client provided the following explanation:

> Intended behavior, design

Description: Throughout multiple contracts, privileges are granted to specific addresses that result in a certain level of centralization of power. Below is a complete list of restricted functions along with the permissions required to call them.

1. `PowerToken`
   1. `onlyStandardGovernor`
   2. `markNextVotingEpochAsActive()`
   3. `markParticipation()`
   4. `setNextCashToken()`
2. `ZeroToken`
   1. `onlyStandardGovernor
   2. `mint()`
3. `ZeroGovernor`
   1. `onlySelf`
   2. `resetToPowerHolders()`
   3. `resetToZeroHolders()`
   4. `setCashToken()`
   5. `setEmergencyProposalThresholdRatio()`
   6. `setZeroProposalThresholdRatio()`
4. `StandardGovernor`
   1. `onlyZeroGovernor`
   2. `setCashToken()`
   3. `onlySelf`
   4. `addToList()`
   5. `removeFromList()`
   6. `removeFromAndAddToList()`
   7. `setKey()`
   8. `onlySelfOrEmergencyGovernor`
   9. `setProposalFee()`
5. `EmergencyGovernor`
   1. `onlySelf`
   2. `addToList()`
   3. `removeFromList()`
   4. `removeFromAndAddToList()
   5. `setKey()`
   6. `setStandardProposalFee()`
   7. `onlyZeroGovernor`
   8. `setThresholdRatio()`
6. `MinterGateway`
   1. `onlyActiveMinter()`
   2. `updateCollateral()`
   3. `proposeRetrieval()`
   4. `proposeMint()`
   5. `mintM()`
   6. `onlyApprovedValidator()`
   7. `cancelMint()`
   8. `freezeMinter()`
   9. `onlyUnfrozenMinter()`
   10. `proposeMint()`
   11. `mintM()`
7. `MToken`
   1. `onlyMinterGateway()`
   2. `mint()`
   3. `burn()`

We would like to note in particular the great power given to validators through their permissions in `MinterGateway`.

Recommendation: The permissions listed above should be clarified and justified to users through public-facing documentation.

MZ-5 Overflow Risks in Unchecked Arithmetic Operations

Low · Mitigated

> **Update**

Marked as "Fixed" by the client.
Addressed in: `0c86903456266e3814d5086c63b680063d03bdb6`.

We found that the following approaches were taken towards the sub-issues:

1. Fixed
2. Acknowledged
3. Acknowledged.
4. Fixed

File(s) affected: `MinterGateway.sol`, `MToken.sol`, `EpochBasedInflationaryVoteToken.sol`, `BatchGovernor.sol`, `PowerToken.sol`

Description: The protocol extensively utilizes unchecked arithmetic operations for efficiency. However, certain functions have been identified where overflow might occur, potentially leading to incorrect calculations or unintended consequences.

The following functions contain the risk of overflow:

1. In the function `MinterGateway.maxAllowedActiveOwedMOf()`, the multiplication of `collateralOf(minter_)` by `mintRatio()` is unchecked. While `mintRatio()` is capped, the product can theoretically exceed the maximum `uint256` value, especially when `collateralOf` returns a high `uint240` value.
2. In the function `ContinuousIndexingMath.divideDown()`, the multiplication with `EXP_SCALED_ONE` can overflow.
3. In the function `ContinuousIndexingMath.divideUp()`, the multiplication with `EXP_SCALED_ONE` can overflow.
4. In the function `ContinuousIndexingMath.multiplyIndices()`, the multiplication can overflow.

While some of the above instances are very unlikely to lead to overflow, we note their existence for the sake of disclosure.

Recommendation: Implement boundary conditions to ensure calculations do not exceed the safe limits of the data types used.

MZ-6 Precision Loss in Token Distribution Affects Small Holders

Low · Mitigated

Update

Marked as "Mitigated" by the client.
Addressed in: `3ae0bd3eda9d0399074549bdd9cdfe5e2f064ebf`.
The client provided the following explanation:

> Granularity was added to the vault

File(s) affected: `PowerToken.sol`

Description: The Power token's design, featuring zero decimal places, heightens precision loss due to integer division in the `_getBootstrapBalance()` function, omitting fractional tokens in distribution. This primarily impacts small stakeholders (e.g., 0.009%), leading to zero allocations after a reset. The extent of this issue's impact is directly proportional to the percentage of the total token supply held by these small stakeholders. Consequently, part of the initial supply remains undistributed, causing a discrepancy between the total holder balances and the Power token's total supply. This issue not only affects distribution precision but also influences governance dynamics. Since governance decisions rely on a simple majority of power token holders, the absence of tokens for small stakeholders post-reset excludes them from governance, potentially centralizing decision-making power. Furthermore, the token loss from small holders leads to a consistent surplus in Power token auctions, as the amount available for auction is determined by the difference between the target and the current total supply. The target token supply is calculated based on the `INITIAL_SUPPLY` and the current token supply is reduced by the unallocated tokens from these holders.

Recommendation: To address the precision loss and its impact on small holders during the Power token reset process, it is recommended to:

1. **Review Distribution Parameters**: Assess the current token distribution parameters, especially those affecting small stakeholders. Understand the trade-offs involved in the design choices, such as the use of zero decimal places.
2. **Document the Impact**: Provide clear, comprehensive documentation detailing how the reset process affects token holders, with particular emphasis on those with small stakes. Ensure that the implications of precision loss and the resulting distribution outcomes are transparently communicated.
3. **Consider Design Adjustments**: Evaluate the feasibility and potential benefits of introducing more decimal places to the Power token. Adding decimal support could allow for a more accurate representation of small stakes and a more equitable distribution of tokens.

MZ-7 Power Token Supply Can Be Bypassed Through Re-Entrancy

Low | Fixed

Update

Marked as "Fixed" by the client.
Addressed in: `28913e3f94217827fb4f568791e8ceb5226052d9` .

The operations were re-ordered.

File(s) affected: `PowerToken.sol`

Description: The `buy()` function of the contract presents a reentrancy vulnerability due to its operational sequence. The issue arises because the function performs a token transfer before completing all state updates, specifically the minting of Power tokens. The `amountToAuction()` function determines the quantity of tokens available based on the current epoch token supply, but the minting in `buy()` happens after the external transfer. This order allows for a possible reentrant call to the `buy()` and allows an attacker to purchase more than the `amount` available during an auction.

Recommendation: While the protocol's whitepaper specifies that the accepted tokens are currently limited to WETH or M tokens, which are not subject to this vulnerability, it is important to note that the protocol's governance has the authority to approve additional tokens. Future token approvals could potentially introduce tokens that are susceptible to reentrancy risks. With this in mind, the following steps are recommended to mitigate potential vulnerabilities:

1. **Reorder Operations in** `buy()` **Function**: Implement the Checks-Effects-Interactions pattern within the `buy()` function. All state changes, especially token minting, should be completed before any external calls or token transfers. This reordering is critical to prevent potential reentrancy.
2. **Use Reentrancy Guard**: Introduce a reentrancy guard in the `buy()` function. This safeguard will block concurrent function calls, significantly mitigating the risk of reentrancy exploits, particularly with tokens like ERC777 or those with similar callback functionalities.

MZ-8 Ambiguous Error Messaging in `MinterGateway.proposeMint()`

Informational | Acknowledged

Update

Marked as "Acknowledged" by the client.
The client provided the following explanation:

> Design

File(s) affected: `MinterGateway.sol`

Description: In the `proposeMint()` function, if the requested mint amount is undercollateralized or the collateral data is stale, the function fails with the `Undercollateralized` error. The same error for both scenarios might not provide sufficient information for the minter to discern whether the issue is due to actual undercollateralization or stale collateral data.

Recommendation: Consider adding distinct error messages for undercollateralization and stale collateral scenarios within the `proposeMint()` function to provide more precise feedback to the minter.

MZ-9 Signatures Missing Expiry

Low | Acknowledged

Update

Marked as "Acknowledged" by the client.
The client provided the following explanation:

> We are following the OZ Governor standard: https://github.com/OpenZeppelin/openzeppelin-contracts/blob/bd325d56b4c62c9c5c1aff048c37c6bb18ac0290/contracts/governance/IGovernor.sol#L291

File(s) affected: `BatchGovernor.sol`

Description: At `castVoteBySig()` and `castVotesBySig()`, signatures are used to cast votes in polls but the signed messages do not come with expiries. As a result, it is not possible for a signer to prevent their signed vote from being deemed invalid at some point in time between the end of the poll and the signing of the signature.

Recommendation: Consider adding a deadline to signed vote casting operations.

MZ-10
Functions for Getting Vote Token Delegatees and Vote Power Amounts May Revert if Gas Too High

Low | Acknowledged

> **Update**
> Marked as "Acknowledged" by the client.
> The client provided the following explanation:
>
> Design

File(s) affected: `EpochBasedVoteToken.sol`, `EpochBasedInflationaryVoteToken.sol`

Description: The functions of `EpochBasedVoteToken`:

1. `pastBalanceOf()`,
2. `pastDelegates()`,
3. `getPastVotes()`, and
4. `pastTotalSupply()`,

together with `EpochBasedInflationaryVoteToken.hasParticipatedAt()`, are designed to determine historical values represented via arrays. In order to do this, they loop through the relevant array from the end, index by index, until the chosen epoch is reached. Given a long enough array, this may result in transaction reversion due to an out-of-gas error. This would amount to certain values no longer being directly reachable via external calls from other contracts.

Recommendation: This issue occurs primarily due to the use of a linear search through the relevant arrays. Use a binary search instead in order to arrive at the entry for the correct epoch.

MZ-11 Missing Input Validation

Low | Mitigated

> **Update**
> Marked as "Fixed" by the client.
> Addressed in: `a7925aa8cd5b4b36fd9f5c35cb947325d206f325`, `0c86903456266e3814d5086c63b680063d03bdb6`, `0da7d2fecff44dc6d9d4c3d9b3cf6b8dd8926f9d`.
>
> We found that the following approaches were taken to the sub-issues:
>
> 1. Fixed
> 2. Fixed
> 3. Acknowledged
> 4. Mitigated
>   1. Acknowledged
>   2. Fixed

File(s) affected: `StandardGovernor.sol`, `ZeroToken.sol`, `SignatureChecker.sol`, `ERC20Extended.sol`, `MinterGateway.sol`

Related Issue(s): SWC-123

Description: It is important to validate inputs, even if they only come from trusted addresses, to avoid human error. Specifically, in the following functions arguments of type `address` may be initialized with value `address(0)`:

1. `ZeroToken`

1. `constructor()`: The addresses in `initialAccounts_` are not checked to be different from `address(0)`.
2. `SignatureChecker`
   1. `decodeECDSASignature()`: Length of `signature` not checked to be `65`.
   2. `SignatureChecker.decodeShortECDSASignature()`: Length of `signature` not checked to be `64`.
3. `ERC20Extended`
   1. `approve()`: `spender_` not checked to be different from `address(0)`.
   2. `permit()`: `spender_` not checked to be different from `address(0)`.
4. `MinterGateway`
   1. `activateMinter()`: `minter_` not checked if already activated, allowing for multiple calls.
   2. `burnM()`: `maxAmount_` (and `maxPrincipalAmount_`) not checked to be different from zero.

Recommendation: We recommend adding the relevant checks.

MZ-12
Rounding Mismatch in M Token Minting and Debt Calculation

Informational | Acknowledged

Update

Marked as "Acknowledged" by the client.
The client provided the following explanation:

> We acknowledge 1 unit discrepancy that can occur for the minter. It is the result of rounding up, down in favor of protocol and storing owed M in form of principal

File(s) affected: `MinterGateway.sol`, `MToken.sol`

Description: The `MinterGateway` contract introduces a rounding discrepancy between the calculation of M tokens owed and the number of M tokens minted. Specifically, while the protocol rounds up the principal amount to determine the M tokens owed, it rounds down during the minting process. This can result in minters accruing debt for M tokens that are not minted.

Recommendation: Consider adjusting the minting process for M Tokens to ensure that the principal amount minted is greater than zero.

MZ-13
Risks in Validator-Based Collateral Attestation with Off-Chain Dependencies

Informational | Acknowledged

Update

Marked as "Acknowledged" by the client.
The client provided the following explanation:

> Design

File(s) affected: `MinterGateway.sol`

Description: The protocol's method for updating on-chain Collateral Value, crucial for M token generation, is contingent on validators' attestation of off-chain data. This reliance introduces potential risks due to dependencies on the accuracy of off-chain data and external validation systems. Moreover, the lack of on-chain incentives for validators, who operate based on off-chain agreements, may influence the reliability of the validation process.

Recommendation: Given the significant privileges of validators, governance participants should exercise great caution when adding validators.

MZ-14
Risk of Standard Governance Griefing Through Excessive Low-Cost Proposals

Informational | Acknowledged

Update

Marked as "Acknowledged" by the client.
The client provided the following explanation:

> Design

Description: The standard governance permits any user to submit proposals with a refundable fee. The `Zero Governance` control permits this fee to be set to zero or a very low value. While this inclusivity is beneficial for broad participation, it opens the door to governance griefing. Malicious actors can exploit the low barrier to entry by flooding the governance with spam proposals. This can overwhelm token holders, who must review and respond to each proposal to avoid dilution of their power tokens.

Recommendation: We propose the following recommendations:

1. **Documenting Risks in Code:**
   - Explicitly document the risks associated with setting the proposal fee to zero. This documentation should highlight the potential for governance griefing and the associated consequences, providing clear guidance for future maintainers or decision-makers.
2. **Preventing Zero Fee Configuration:**
   - Modify the `ZeroGovernor.setCashToken()` function to prevent the proposal fee from being set to zero. Implement a safeguard in the code that enforces a sufficiently high minimum fee threshold, ensuring that there is always some cost associated with submitting a proposal.

MZ-15
Limitation of Padé Approximant in Approximating Exponential Functions for Financial Calculations

Informational | Acknowledged

Update

Marked as "Acknowledged" by the client.
The client provided the following explanation:

> Padé was chosen because it is accurate enough while being cheaper, but more importantly, never exploding/overflowing/reverting for all inputs, since the resulting value plateaus (before dropping back down).

File(s) affected: `ContinuousIndexingMath.sol`

Description: The use of the Padé approximant to approximate the exponential function e^{rt} in compound interest calculations may lead to accuracy limitations. This approximant, while efficient in certain scenarios, can produce discrepancies in financial calculations when used for certain values of rt. For instance, at $rt = 2.4$, the approximation starts to show a noticeable deviation of approximately 1 basis point from the actual value of e^{rt}. It is important to note that the reasonable parameters provided in the protocol parameter specification, calculating compound interest with an APY of 400% over 30 days (equivalent to $rt \approx 0.3288$), the approximation remains relatively accurate. However, with significant APYs, the protocol must exercise caution. It is recommended to model the compounding effect with the new APY to ensure the protocol can handle the computations with sufficient accuracy, avoiding significant inaccuracies in financial calculations for different values of rt.

Recommendation: We propose the consideration of the following recommendations:

1. **Awareness of Limitations**: Users and developers should be cognizant of the range within which the Padé approximant remains accurate.
2. **Documentation Update**: The limitations of the Padé approximant, particularly its valid range of rt values, should be explicitly documented in the code. This documentation will help ensure that any future modifications or different use cases are aware of these limitations.

MZ-16 Loss of Precision Due to Division before Multiplication

Informational | Fixed

Update

Marked as "Fixed" by the client.
Addressed in: `cac3c24b038d9d6c71bad1132b9f4617e91ef2f1` .

File(s) affected: `StableEarnerRateModel.sol`

Description: Division before multiplication may result in a loss of precision when the operations are carried over integer numbers. This occurs at `StableEarnerRateModel.sol#L106` :

```
    int256 lnArg_ = int256(
        1e12 + ((((uint256(totalActiveOwedM_) * (deltaMinterIndex_ - 1e12)) / 1e12) * 1e12) /
totalEarningSupply_)
    );
```

Recommendation: Rewrite equations so that division happens after multiplication.

MZ-17 Clone-and-Own

Informational | Acknowledged

Update

Marked as "Acknowledged" by the client.
The client provided the following explanation:

> We made the deliberate approach to create our own contracts that we can extend with the latest EIPs instead of relying on third party libraries. Also, it allows us to have uniform and detailed error messages, instead of an uncontrolled mix of requires, boolean returns, and reverts.

File(s) affected: `SignatureChecker.sol`, `UIntMath.sol`, `ERC712.sol`, `ERC20Extended.sol`

Description: The clone-and-own approach involves copying and adjusting open source code at one's own discretion. From the development perspective, it is initially beneficial as it reduces the amount of effort. However, from the security perspective, it involves some risks as the code may not follow the best practices, may contain a security vulnerability, or may include intentionally or unintentionally modified upstream libraries.

The following instances have been identified:

1. `SignatureChecker.sol`: OpenZeppelin `ECDSA.sol` and `SignatureChecker.sol`.
2. `UIntMath.sol`: OpenZeppelin `SafeCast.sol`.
3. `ERC712.sol`: OpenZeppelin `EIP712.sol`.
4. `ERC20Extended.sol`: OpenZeppelin `ERC20.sol` and `ERC20Permit.sol`.

Recommendation: Rather than the clone-and-own approach, a good industry practice is to use a package manager (e.g., npm) for handling library dependencies. This eliminates the clone-and-own risks yet allows for following best practices, such as, using libraries. If the file is cloned anyway, a comment including the repository, commit hash of the version cloned, and the summary of modifications (if any) should be added. This helps to improve traceability of the file.

MZ-18 Code Documentation

Informational | Fixed

Update

Marked as "Fixed" by the client.
Addressed in: `5445cff372e13798b31969ed5a35fcfb40fea9ea`, `829db205a5b716ec80059aa5451a03a4c6422ce5`, `0a0cae40c2c88625cb455fd41bb2a5740f85a7d3`.

Description: We recognized a few places where the code documentation can be improved:

1. `TTGRegistrarReader.sol#L24`: The comment should say `earners ignore list` rather than `earners list`.
2. `MinterGateway.sol#L98–99`: NatSpec comments similar, consider removing one.
3. `ThresholdGovernor.sol#L16`: Link is outdated and should rather be https://portal.thirdweb.com/contracts/build/base-contracts/erc-20/vote.

The following typos were also spotted:

1. `ContractHelper.sol#L6`: `aan` → `an`.
2. `ContinuousIndexingMath.sol#L88`: `costs` → `cost`.
3. `MinterGateway.sol#L1032`: `BY` → `By`.
4. `EpochBasedVoteToken.sol#L285`: `array of given by` → `by`.
5. `EpochBasedInflationaryVoteToken.sol#L12`: `,a nd` → `, and`.
6. `EpochBasedInflationaryVoteToken.sol#L116`: `the its` → `its`.
7. `PowerToken.sol#L309` and `L312`: `that in` → `that is`.

Recommendation: Consider correcting the above issues.

MZ-19 Undocumented Magic Constants

Informational | Fixed

Update

Marked as "Fixed" by the client.

Addressed in: `fa751d7bf476d738a6b010a46da8afe772b16b5b`.

File(s) affected: `StableEarnerRateModel.sol`

Description: To improve readability and lower the risk of introducing errors when making code changes, it is advised to not use magic constants throughout code, but instead declare them once (as constant and commented) and use these constant variables instead. The following instances should therefore be changed accordingly:

1. `StableEarnerRateModel.sol#L106` : `1e12` .
2. `StableEarnerRateModel.sol#L109` : `1e6` .

Recommendation: Ensure that all constants are defined as intended, and use named constants where appropriate. Add documentation explaining the rationale behind each constant.

MZ-20 Outstanding Todo Comments

Informational | Fixed

Update

Marked as "Fixed" by the client.

Addressed in: `ead0c859f9f61d3347ad954288d0d15c268fa07c`.

File(s) affected: `BatchGovernor.sol` , `ThresholdGovernor.sol`

Description: Before rolling out code in production, any pending `TODO` items in code should be resolved in order to not deploy potentially unfinished code. In this regard the following `TODO` items still remain in code and should be resolved:

1. `BatchGovernor.sol#L303` : "TODO: Check if ignoring the voter's reason breaks community compatibility of this event."
2. `ThresholdGovernor.sol#L12` : "TODO: Determine `quorumNumerator` / `quorumDenominator` / `QuorumNumeratorUpdated` stuff, and how it applies to tokens"

Recommendation: Resolve the `TODO` items.

MZ-21 Adherence to Specification

Informational | Acknowledged

Update

Marked as "Fixed" by the client.

Addressed in: `5828c44739fac3f7799171072410aad1ca6c8d96`.

The client has responded to the issue to some extent in the code; however, we have not yet received the new specification.

Description: We identified a number of occurrences where the code does not match the specification provided.

1. Functions `allowEarningOnBehalfOf()` , `disallowEarningOnBehalfOf()` and `stopEarning()` are callable by anyone, not just earners.
2. Function `proposeRetrieval()` : Implemented logic mismatches specification. A check against the re-calculated active owed M, as described in the specification, is not being performed.
3. `activateMinter()` : Function additionally checks that the minter has not been explicitly deactivated via `_minterStates[minter_].isDeactivated` .
4. `deactivateMinter()` : The flag `isActive` is set to `false` , not to `true` .
5. `startEarning()` : Additionally the flag `_balances[account_].isEarning` is set to `true` and `updateIndex()` is called.
6. `stopEarning()` : Additionally the flag `_balances[account_].isEarning` is set to `false` , `updateIndex()` is called and the account opts out of earning of behalf.
7. `_transfer()` : In the case that both, sender and receiver, are of the same kind (both are earning or both are non-earning) the code adds and subtracts the values from the raw balances accordingly and returns early, skipping the call to `updateIndex()` , which is otherwise performed.

Recommendation: Either correct the specification or correct the code to match the specification.

Definitions

- **High severity** – High-severity issues usually put a large number of users' sensitive information at risk, or are reasonably likely to lead to catastrophic impact for client's reputation or serious financial implications for client and users.

- **Medium severity** – Medium-severity issues tend to put a subset of users' sensitive information at risk, would be detrimental for the client's reputation if exploited, or are reasonably likely to lead to moderate financial impact.
- **Low severity** – The risk is relatively small and could not be exploited on a recurring basis, or is a risk that the client has indicated is low impact in view of the client's business circumstances.
- **Informational** – The issue does not post an immediate risk, but is relevant to security best practices or Defence in Depth.
- **Undetermined** – The impact of the issue is uncertain.
- **Fixed** – Adjusted program implementation, requirements or constraints to eliminate the risk.
- **Mitigated** – Implemented actions to minimize the impact or likelihood of the risk.
- **Acknowledged** – The issue remains in the code but is a result of an intentional business or design decision. As such, it is supposed to be addressed outside the programmatic means, such as: 1) comments, documentation, README, FAQ; 2) business processes; 3) analyses showing that the issue shall have no negative consequences in practice (e.g., gas analysis, deployment settings).

Appendix

File Signatures

The following are the SHA-256 hashes of the reviewed files. A file with a different SHA-256 hash has been modified, intentionally or otherwise, after the security review. You are cautioned that a different SHA-256 hash could be (but is not necessarily) an indication of a changed condition or potential vulnerability that was not within the scope of the review.

Files

- `f09...424 ./mzero-contracts/spog/src/Registrar.sol`
- `c2c...d9e ./mzero-contracts/spog/src/PowerTokenDeployer.sol`
- `d67...22e ./mzero-contracts/spog/src/PowerToken.sol`
- `621...9e6 ./mzero-contracts/spog/src/StandardGovernor.sol`
- `134...c73 ./mzero-contracts/spog/src/StandardGovernorDeployer.sol`
- `ed2...48d ./mzero-contracts/spog/src/ZeroToken.sol`
- `bf6...728 ./mzero-contracts/spog/src/ZeroGovernor.sol`
- `c6e...06b ./mzero-contracts/spog/src/DistributionVault.sol`
- `198...923 ./mzero-contracts/spog/src/EmergencyGovernor.sol`
- `263...88e ./mzero-contracts/spog/src/EmergencyGovernorDeployer.sol`
- `275...76f ./mzero-contracts/spog/src/PowerBootstrapToken.sol`
- `9a5...964 ./mzero-contracts/spog/src/interfaces/IDeployer.sol`
- `0d5...0ee ./mzero-contracts/spog/src/interfaces/IStandardGovernorDeployer.sol`
- `75b...fd2 ./mzero-contracts/spog/src/interfaces/IEmergencyGovernor.sol`
- `b33...646 ./mzero-contracts/spog/src/interfaces/IDistributionVault.sol`
- `a65...9e8 ./mzero-contracts/spog/src/interfaces/IPowerBootstrapToken.sol`
- `94d...297 ./mzero-contracts/spog/src/interfaces/IRegistrar.sol`
- `6d5...c62 ./mzero-contracts/spog/src/interfaces/IStandardGovernor.sol`
- `78a...85a ./mzero-contracts/spog/src/interfaces/IZeroGovernor.sol`
- `1ce...735 ./mzero-contracts/spog/src/interfaces/IEmergencyGovernorDeployer.sol`
- `39a...96d ./mzero-contracts/spog/src/interfaces/IZeroToken.sol`
- `dca...a82 ./mzero-contracts/spog/src/interfaces/IPowerTokenDeployer.sol`
- `597...d51 ./mzero-contracts/spog/src/interfaces/IPowerToken.sol`
- `811...d07 ./mzero-contracts/spog/src/libs/PureEpochs.sol`
- `3cf...527 ./mzero-contracts/spog/src/abstract/ERC5805.sol`
- `90b...7c3 ./mzero-contracts/spog/src/abstract/ThresholdGovernor.sol`
- `0fb...a8a ./mzero-contracts/spog/src/abstract/BatchGovernor.sol`
- `fb7...b0d ./mzero-contracts/spog/src/abstract/EpochBasedVoteToken.sol`
- `e14...016 ./mzero-contracts/spog/src/abstract/EpochBasedInflationaryVoteToken.sol`
- `ac5...4d2 ./mzero-contracts/spog/src/abstract/interfaces/IERC5805.sol`
- `efe...8f5 ./mzero-contracts/spog/src/abstract/interfaces/IERC6372.sol`
- `14a...f4c ./mzero-contracts/spog/src/abstract/interfaces/IEpochBasedVoteToken.sol`
- `2ec...99a ./mzero-contracts/spog/src/abstract/interfaces/IGovernor.sol`
- `fd4...0b0 ./mzero-contracts/spog/src/abstract/interfaces/IBatchGovernor.sol`
- `938...e5a ./mzero-contracts/spog/src/abstract/interfaces/IThresholdGovernor.sol`

- f7b...8bc ./mzero-contracts/spog/src/abstract/interfaces/IEpochBasedInflationaryVoteToken.sol
- 1fb...0f6 ./mzero-contracts/common/src/ERC3009.sol
- 9f3...d2b ./mzero-contracts/common/src/ERC20Extended.sol
- 57b...9f2 ./mzero-contracts/common/src/ERC712.sol
- 986...928 ./mzero-contracts/common/src/ContractHelper.sol
- 4c3...c3f ./mzero-contracts/common/src/StatefulERC712.sol
- 21d...71a ./mzero-contracts/common/src/interfaces/IERC20.sol
- 5bf...69c ./mzero-contracts/common/src/interfaces/IERC20Extended.sol
- d36...e04 ./mzero-contracts/common/src/interfaces/IERC3009.sol
- 350...415 ./mzero-contracts/common/src/interfaces/IERC712.sol
- 963...8b7 ./mzero-contracts/common/src/interfaces/IStatefulERC712.sol
- eca...b65 ./mzero-contracts/common/src/interfaces/IERC1271.sol
- 6d2...655 ./mzero-contracts/common/src/libs/UIntMath.sol
- 2c1...4a7 ./mzero-contracts/common/src/libs/SignatureChecker.sol
- 467...b8f ./mzero-contracts/protocol/src/MinterGateway.sol
- 668...b7c ./mzero-contracts/protocol/src/MToken.sol
- cea...4d1 ./mzero-contracts/protocol/src/interfaces/IContinuousIndexing.sol
- 83e...1f2 ./mzero-contracts/protocol/src/interfaces/ITTGRegistrar.sol
- a96...db4 ./mzero-contracts/protocol/src/interfaces/IRateModel.sol
- de4...8ee ./mzero-contracts/protocol/src/interfaces/IMinterGateway.sol
- 1ac...b3d ./mzero-contracts/protocol/src/interfaces/IMToken.sol
- a47...6b8 ./mzero-contracts/protocol/src/rateModels/SplitEarnerRateModel.sol
- 107...50f ./mzero-contracts/protocol/src/rateModels/StableEarnerRateModel.sol
- 15c...7b9 ./mzero-contracts/protocol/src/rateModels/MinterRateModel.sol
- b72...363 ./mzero-contracts/protocol/src/rateModels/interfaces/IEarnerRateModel.sol
- 1cd...38f ./mzero-contracts/protocol/src/rateModels/interfaces/IStableEarnerRateModel.sol
- 954...f1b ./mzero-contracts/protocol/src/rateModels/interfaces/IMinterRateModel.sol
- da6...74d ./mzero-contracts/protocol/src/libs/TTGRegistrarReader.sol
- 13d...2e8 ./mzero-contracts/protocol/src/libs/ContinuousIndexingMath.sol
- dc4...822 ./mzero-contracts/protocol/src/abstract/ContinuousIndexing.sol

Tests

- 173...c18 ./mzero-tests/spog/test/EpochBasedInflationaryVoteToken.t.sol
- 3da...ab5 ./mzero-tests/spog/test/StandardGovernor.t.sol
- 060...d97 ./mzero-tests/spog/test/DistributionVault.t.sol
- f4b...dba ./mzero-tests/spog/test/PowerTokenDeployer.t.sol
- b80...2e5 ./mzero-tests/spog/test/Registrar.t.sol
- 055...577 ./mzero-tests/spog/test/PureEpochs.t.sol
- 0e1...427 ./mzero-tests/spog/test/ZeroToken.t.sol
- ab7...4cd ./mzero-tests/spog/test/EpochBasedVoteToken.t.sol
- 54a...b4f ./mzero-tests/spog/test/ZeroGovernor.t.sol
- 8d2...fc6 ./mzero-tests/spog/test/PowerToken.t.sol
- a04...26b ./mzero-tests/spog/test/EmergencyGovernor.t.sol
- 463...d9a ./mzero-tests/spog/test/EmergencyGovernorDeployer.t.sol
- 3ff...a0d ./mzero-tests/spog/test/StandardGovernorDeployer.t.sol
- c89...0ff ./mzero-tests/spog/test/fuzz/EpochBasedInflationaryVoteTokenFuzz.t.sol
- 4cc...e14 ./mzero-tests/spog/test/utils/TestUtils.sol
- 819...a68 ./mzero-tests/spog/test/utils/Mocks.sol
- bf0...894 ./mzero-tests/spog/test/utils/ERC20ExtendedHarness.sol
- ef2...81d ./mzero-tests/spog/test/utils/EmergencyGovernorHarness.sol
- 45c...a15 ./mzero-tests/spog/test/utils/EpochBasedInflationaryVoteTokenHarness.sol
- c96...67e ./mzero-tests/spog/test/utils/Invariants.sol
- d11...158 ./mzero-tests/spog/test/utils/ZeroTokenHarness.sol
- baf...3c7 ./mzero-tests/spog/test/utils/PowerTokenHarness.sol
- eba...fd0 ./mzero-tests/spog/test/utils/ZeroGovernorHarness.sol

- `41c...30b ./mzero-tests/spog/test/utils/StandardGovernorHarness.sol`
- `3c2...0b9 ./mzero-tests/spog/test/utils/EpochBasedVoteTokenHarness.sol`
- `30d...672 ./mzero-tests/spog/test/integration/IntegrationBaseSetup.t.sol`
- `565...56a ./mzero-tests/spog/test/integration/Integration.t.sol`
- `c92...e23 ./mzero-tests/spog/test/integration/vault/distributionVault.t.sol`
- `8b6...7f9 ./mzero-tests/spog/test/integration/emergency-governor/propose/emergencyGovernorPropose.t.sol`
- `fb0...81a ./mzero-tests/spog/test/integration/standard-governor/propose/standardGovernorPropose.t.sol`
- `156...21e ./mzero-tests/spog/test/integration/inflation-rewards/powerInflationZeroRewards.t.sol`
- `901...d05 ./mzero-tests/spog/test/integration/zero-governor/set-thresholds/setZeroEmergencyThresholds.t.sol`
- `015...523 ./mzero-tests/spog/test/integration/zero-governor/set-cash-token/setCashToken.t.sol`
- `1e4...2ff ./mzero-tests/spog/test/integration/zero-governor/reset/ResetIntegrationBaseSetup.t.sol`
- `395...785 ./mzero-tests/spog/test/integration/zero-governor/reset/reset-to-power-holders/resetToPowerHolders.t.sol`
- `c10...ab9 ./mzero-tests/spog/test/integration/zero-governor/reset/reset-to-zero-holders/resetToZeroHolders.t.sol`
- `6a1...7ee ./mzero-tests/spog/test/integration/zero-governor/propose/zeroGovernorPropose.t.sol`
- `a59...b24 ./mzero-tests/spog/test/integration/auction/auction.t.sol`
- `d2b...0a7 ./mzero-tests/common/test/ContractHelper.t.sol`
- `432...8a3 ./mzero-tests/common/test/ERC3009.t.sol`
- `b3f...77e ./mzero-tests/common/test/SignatureChecker.t.sol`
- `5cd...39f ./mzero-tests/common/test/UIntMath.t.sol`
- `2ac...22a ./mzero-tests/common/test/utils/ERC20ExtendedHarness.sol`
- `2e1...1b2 ./mzero-tests/common/test/utils/ContractHelperHarness.sol`
- `101...42a ./mzero-tests/common/test/utils/UIntMathHarness.sol`
- `50f...0f0 ./mzero-tests/common/test/utils/TestUtils.t.sol`
- `bf6...065 ./mzero-tests/common/test/utils/SignatureCheckerHarness.sol`
- `c47...321 ./mzero-tests/protocol/test/ContinuousIndexingMath.t.sol`
- `3b8...a5a ./mzero-tests/protocol/test/MToken.t.sol`
- `108...f98 ./mzero-tests/protocol/test/MinterGateway.t.sol`
- `659...686 ./mzero-tests/protocol/test/RateModel.t.sol`
- `01f...d80 ./mzero-tests/protocol/test/fuzz/Fuzz.t.sol`
- `9cc...290 ./mzero-tests/protocol/test/utils/TestUtils.sol`
- `20c...a24 ./mzero-tests/protocol/test/utils/MinterGatewayHarness.sol`
- `3e1...91b ./mzero-tests/protocol/test/utils/ContinuousIndexingMathHarness.sol`
- `627...958 ./mzero-tests/protocol/test/utils/Mocks.sol`
- `cda...edf ./mzero-tests/protocol/test/utils/MTokenHarness.sol`
- `583...bf8 ./mzero-tests/protocol/test/utils/DigestHelper.sol`
- `e9f...3ab ./mzero-tests/protocol/test/integration/IntegrationBaseSetup.t.sol`
- `49a...060 ./mzero-tests/protocol/test/integration/Integration.t.sol`
- `a0f...f35 ./mzero-tests/protocol/test/integration/minter-gateway/Integration.t.sol`
- `f4d...8c9 ./mzero-tests/protocol/test/integration/minter-gateway/burn-m/burnM.t.sol`
- `e1e...d43 ./mzero-tests/protocol/test/integration/minter-gateway/update-collateral/updateCollateral.t.sol`
- `b6d...e8d ./mzero-tests/protocol/test/integration/minter-gateway/deactivate-minter/deactivateMinter.t.sol`
- `dca...359 ./mzero-tests/protocol/test/invariant/Invariant.t.sol`

Toolset

The notes below outline the setup and steps performed in the process of this audit.

Setup

Tool Setup:

- Slither v0.10.0

Steps taken to run the tools:

1. Install the Slither tool: `pip3 install slither-analyzer`
2. Run Slither from the project directory: `slither .`

Automated Analysis

Slither

Relevant findings from Slither have been included in the report.

Test Suite Results

The test suites are robust, containing fuzzing and integration tests in addition to unit tests. We would, however, like to see integration testing that involves the use of both governance and main protocol contracts.

```
MZero-Labs/common
Running 1 test for test/ContractHelper.t.sol:ContractHelperTests
[PASS] test_full() (gas: 661461)
Test result: ok. 1 passed; 0 failed; 0 skipped; finished in 1.37ms

Running 16 tests for test/UIntMath.t.sol:UIntMathTests
[PASS] test_bound112() (gas: 6595)
[PASS] test_bound240() (gas: 6551)
[PASS] test_bound32() (gas: 6531)
[PASS] test_max40() (gas: 9059)
[PASS] test_min112() (gas: 8951)
[PASS] test_min240() (gas: 9150)
[PASS] test_min256() (gas: 9018)
[PASS] test_min32() (gas: 9127)
[PASS] test_min40() (gas: 9081)
[PASS] test_min40IgnoreZero() (gas: 15696)
[PASS] test_safe112() (gas: 10840)
[PASS] test_safe128() (gas: 10775)
[PASS] test_safe16() (gas: 10906)
[PASS] test_safe240() (gas: 10860)
[PASS] test_safe40() (gas: 10815)
[PASS] test_safe48() (gas: 10817)
Test result: ok. 16 passed; 0 failed; 0 skipped; finished in 1.66ms

Running 38 tests for test/SignatureChecker.t.sol:SignatureCheckerTests
[PASS] test_decodeECDSASignature() (gas: 8811)
[PASS] test_isValidECDSASignature_bytes() (gas: 19091)
[PASS] test_isValidECDSASignature_bytes_invalid() (gas: 45255)
[PASS] test_isValidECDSASignature_rvs() (gas: 17675)
[PASS] test_isValidECDSASignature_rvs_invalid() (gas: 41451)
[PASS] test_isValidECDSASignature_vrs() (gas: 17746)
[PASS] test_isValidECDSASignature_vrs_invalid() (gas: 38640)
[PASS] test_isValidERC1271Signature() (gas: 192279)
[PASS] test_isValidERC1271Signature_accountFailsSilently() (gas: 190285)
[PASS] test_isValidERC1271Signature_accountReturnsInvalidData() (gas: 192245)
[PASS] test_isValidERC1271Signature_accountReturnsNothing() (gas: 167678)
[PASS] test_isValidERC1271Signature_accountReturnsTrue() (gas: 184671)
[PASS] test_isValidERC1271Signature_accountReverts() (gas: 189134)
[PASS] test_isValidERC1271Signature_accountWithFallback() (gas: 55495)
[PASS] test_isValidERC1271Signature_accountWithoutFallback() (gas: 53765)
[PASS] test_isValidERC1271Signature_emptyAccount() (gas: 16338)
[PASS] test_isValidSignature_ecdsa() (gas: 27452)
[PASS] test_isValidSignature_erc1271() (gas: 192755)
[PASS] test_isValidSignature_invalid() (gas: 1110911)
[PASS] test_recoverECDSASigner_bytes() (gas: 18885)
[PASS] test_recoverECDSASigner_bytes_invalidSignature() (gas: 13050)
[PASS] test_recoverECDSASigner_bytes_invalidSignatureS() (gas: 9532)
[PASS] test_recoverECDSASigner_bytes_invalidSignatureV() (gas: 9354)
[PASS] test_recoverECDSASigner_rvs() (gas: 17417)
[PASS] test_recoverECDSASigner_rvs_invalidSignature() (gas: 11733)
[PASS] test_recoverECDSASigner_rvs_invalidSignatureS() (gas: 8064)
[PASS] test_recoverECDSASigner_vrs() (gas: 17567)
[PASS] test_recoverECDSASigner_vrs_invalidSignature() (gas: 12161)
[PASS] test_recoverECDSASigner_vrs_invalidSignatureS() (gas: 8427)
[PASS] test_recoverECDSASigner_vrs_invalidSignatureV() (gas: 8420)
[PASS] test_validateECDSASignature_bytes() (gas: 19047)
[PASS] test_validateECDSASignature_bytes_invalid() (gas: 44519)
```

```
[PASS] test_validateECDSASignature_rvs() (gas: 17654)
[PASS] test_validateECDSASignature_rvs_invalid() (gas: 37656)
[PASS] test_validateECDSASignature_vrs() (gas: 17653)
[PASS] test_validateECDSASignature_vrs_invalid() (gas: 38401)
[PASS] test_validateRecoveredSigner() (gas: 6852)
[PASS] test_validateRecoveredSigner_mismatch() (gas: 6951)
Test result: ok. 38 passed; 0 failed; 0 skipped; finished in 6.98ms

Running 27 tests for test/ERC3009.t.sol:ERC3009Tests
[PASS] test_authorizationState() (gas: 28047)
[PASS] test_cancelAuthorizationTypehash() (gas: 5847)
[PASS] test_cancelAuthorization_authorizationAlreadyCanceled() (gas: 53406)
[PASS] test_cancelAuthorization_cancelTransferAuthorization_fullSignature() (gas: 56860)
[PASS] test_cancelAuthorization_cancelTransferAuthorization_rvsSignature() (gas: 55382)
[PASS] test_cancelAuthorization_cancelTransferAuthorization_vrsSignature() (gas: 55500)
[PASS] test_receiveWithAuthorizationTypehash() (gas: 5804)
[PASS] test_receiveWithAuthorization_authorizationAlreadyUsed() (gas: 56900)
[PASS] test_receiveWithAuthorization_authorizationExpired() (gas: 32211)
[PASS] test_receiveWithAuthorization_authorizationNotYetValid() (gas: 32191)
[PASS] test_receiveWithAuthorization_callerMustBePayee() (gas: 34388)
[PASS] test_receiveWithAuthorization_cannotUseTransferAuthorization() (gas: 30953)
[PASS] test_receiveWithAuthorization_fullSignature() (gas: 62763)
[PASS] test_receiveWithAuthorization_invalidParameter() (gas: 87691)
[PASS] test_receiveWithAuthorization_invalidSigner() (gas: 30955)
[PASS] test_receiveWithAuthorization_rvsSignature() (gas: 61299)
[PASS] test_receiveWithAuthorization_vrsSignature() (gas: 61326)
[PASS] test_transferWithAuthorizationTypehash() (gas: 5869)
[PASS] test_transferWithAuthorization_authorizationAlreadyUsed() (gas: 58831)
[PASS] test_transferWithAuthorization_authorizationExpired() (gas: 34163)
[PASS] test_transferWithAuthorization_authorizationNotYetValid() (gas: 34103)
[PASS] test_transferWithAuthorization_cannotUseReceiveAuthorization() (gas: 32933)
[PASS] test_transferWithAuthorization_fullSignature() (gas: 64704)
[PASS] test_transferWithAuthorization_invalidParameter() (gas: 89612)
[PASS] test_transferWithAuthorization_invalidSigner() (gas: 32887)
[PASS] test_transferWithAuthorization_rvsSignature() (gas: 63172)
[PASS] test_transferWithAuthorization_vrsSignature() (gas: 63214)
Test result: ok. 27 passed; 0 failed; 0 skipped; finished in 6.88ms

Ran 4 test suites: 82 tests passed, 0 failed, 0 skipped (82 total tests)

MZero-Labs/protocol
Running 1 test for test/RateModel.t.sol:ContinuousIndexingMathTests
[PASS] test_stableModel_getSafeEarnerRate() (gas: 54075)
Test result: ok. 1 passed; 0 failed; 0 skipped; finished in 686.54µs

Running 1 test for test/integration/minter-gateway/burn-m/burnM.t.sol:BurnM_IntegrationTest
[PASS] test_burnM_updateCollateralIntervalChange() (gas: 631412)
Logs:
  deployer:  0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Expected Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  M Token address:  0x8Ad159a275AEE56fb2334DBb69036E9c7baCEe9b
  Earner Rate Model address:  0x9c52B2C4A89E2BE37972d18dA937cbAd8AA8bd50
  Minter Rate Model address:  0xfF2Bd636B9Fc89645C2D336aeaDE2E4AbaFe1eA5

Test result: ok. 1 passed; 0 failed; 0 skipped; finished in 4.97ms

Running 1 test for test/integration/minter-gateway/deactivate-
minter/deactivateMinter.t.sol:DeactivateMinter_IntegrationTest
[PASS] test_deactivateMinter_updateCollateralIntervalChange() (gas: 566807)
Logs:
  deployer:  0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Expected Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  M Token address:  0x8Ad159a275AEE56fb2334DBb69036E9c7baCEe9b
  Earner Rate Model address:  0x9c52B2C4A89E2BE37972d18dA937cbAd8AA8bd50
  Minter Rate Model address:  0xfF2Bd636B9Fc89645C2D336aeaDE2E4AbaFe1eA5

Test result: ok. 1 passed; 0 failed; 0 skipped; finished in 8.29ms

Running 1 test for test/integration/Integration.t.sol:IntegrationTests
[SKIP] test_story1() (gas: 0)
```

```
Test result: ok. 0 passed; 0 failed; 1 skipped; finished in 9.53ms

Running 3 tests for test/integration/minter-gateway/update-
collateral/updateCollateral.t.sol:UpdateCollateral_IntegrationTest
[PASS] test_updateCollateral_mintRatioChange() (gas: 601259)
Logs:
  deployer:  0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Expected Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  M Token address:  0x8Ad159a275AEE56fb2334DBb69036E9c7baCEe9b
  Earner Rate Model address:  0x9c52B2C4A89E2BE37972d18dA937cbAd8AA8bd50
  Minter Rate Model address:  0xfF2Bd636B9Fc89645C2D336aeaDE2E4AbaFe1eA5

[PASS] test_updateCollateral_penaltyRateChange() (gas: 699335)
Logs:
  deployer:  0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Expected Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  M Token address:  0x8Ad159a275AEE56fb2334DBb69036E9c7baCEe9b
  Earner Rate Model address:  0x9c52B2C4A89E2BE37972d18dA937cbAd8AA8bd50
  Minter Rate Model address:  0xfF2Bd636B9Fc89645C2D336aeaDE2E4AbaFe1eA5

[PASS] test_updateCollateral_updateCollateralIntervalChange() (gas: 705494)
Logs:
  deployer:  0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Expected Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  M Token address:  0x8Ad159a275AEE56fb2334DBb69036E9c7baCEe9b
  Earner Rate Model address:  0x9c52B2C4A89E2BE37972d18dA937cbAd8AA8bd50
  Minter Rate Model address:  0xfF2Bd636B9Fc89645C2D336aeaDE2E4AbaFe1eA5

Test result: ok. 3 passed; 0 failed; 0 skipped; finished in 17.37ms

Running 74 tests for test/MinterGateway.t.sol:MinterGatewayTests
[PASS] test_activateMinter() (gas: 36536)
[PASS] test_activateMinter_deactivatedMinter() (gas: 27712)
[PASS] test_activateMinter_notApprovedMinter() (gas: 20767)
[PASS] test_activeOwedMOf() (gas: 46926)
[PASS] test_activeOwedM_indexing() (gas: 51307)
[PASS] test_burnM() (gas: 191478)
[PASS] test_burnM_imposePenaltyForExpiredCollateralValue() (gas: 181122)
[PASS] test_burnM_notEnoughBalanceToRepay() (gas: 93491)
[PASS] test_burnM_repayHalfOfOutstandingValue() (gas: 141431)
[PASS] test_cancelMint_byValidator() (gas: 59556)
[PASS] test_cancelMint_invalidMintProposal() (gas: 37126)
[PASS] test_cancelMint_notApprovedValidator() (gas: 21075)
[PASS] test_collateralExpiryTimestampOf() (gas: 57871)
[PASS] test_collateralPenaltyDeadlineOf() (gas: 98450)
[PASS] test_constructor() (gas: 14793)
[PASS] test_constructor_zeroMToken() (gas: 104862)
[PASS] test_constructor_zeroTTGRegistrar() (gas: 99033)
[PASS] test_constructor_zeroTTGVault() (gas: 103715)
[PASS] test_deactivateMinter() (gas: 185737)
[PASS] test_deactivateMinter_alreadyInactiveMinter() (gas: 16590)
[PASS] test_deactivateMinter_imposePenaltyForExpiredCollateralValue() (gas: 151896)
[PASS] test_deactivateMinter_stillApprovedMinter() (gas: 23018)
[PASS] test_emptyRateModel() (gas: 17062)
[PASS] test_freezeMinter() (gas: 170093)
[PASS] test_freezeMinter_notApprovedValidator() (gas: 20660)
[PASS] test_freezeMinter_sequence() (gas: 64514)
[PASS] test_getMissedCollateralUpdateParameters_newMinter() (gas: 7918)
[PASS] test_getMissedCollateralUpdateParameters_zeroNewUpdateInterval() (gas: 8371)
[PASS] test_getPenaltyForMissedCollateralUpdates_moreMissedIntervalsDueToReducedInterval() (gas: 96472)
[PASS] test_getPenaltyForMissedCollateralUpdates_noMissedIntervals() (gas: 80489)
[PASS] test_getPenaltyForMissedCollateralUpdates_oneMissedInterval() (gas: 89106)
[PASS] test_getPenaltyForMissedCollateralUpdates_threeMissedInterval() (gas: 89542)
[PASS] test_getPenaltyForMissedCollateralUpdates_updateCollateralIntervalHasChanged() (gas: 118668)
[PASS] test_imposePenalty_penalizedUntil() (gas: 253081)
[PASS] test_imposePenalty_penalizedUntil_reducedInterval() (gas: 205393)
[PASS] test_imposePenalty_principalOfTotalActiveOwedMOverflows() (gas: 226230)
[PASS] test_inactiveOwedMOf() (gas: 35043)
[PASS] test_mintM() (gas: 177152)
```

```
[PASS] test_mintM_expiredMintRequest() (gas: 82112)
[PASS] test_mintM_frozenMinter() (gas: 53449)
[PASS] test_mintM_inactiveMinter() (gas: 14643)
[PASS] test_mintM_invalidMintRequest() (gas: 21624)
[PASS] test_mintM_invalidMintRequest_mismatchOfIds() (gas: 67946)
[PASS] test_mintM_overflowsPrincipalOfTotalOwedM() (gas: 121221)
[PASS] test_mintM_pendingMintRequest() (gas: 77976)
[PASS] test_mintM_undercollateralizedMint() (gas: 126158)
[PASS] test_mintM_undercollateralizedMint_outdatedCollateral() (gas: 124003)
[PASS] test_principalOfTotalActiveOwedM() (gas: 29822)
[PASS] test_proposeMint() (gas: 143882)
[PASS] test_proposeMint_frozenMinter() (gas: 55933)
[PASS] test_proposeMint_inactiveMinter() (gas: 18676)
[PASS] test_proposeMint_undercollateralizedMint() (gas: 69839)
[PASS] test_proposeRetrieval() (gas: 302441)
[PASS] test_proposeRetrieval_RetrievalsExceedCollateral() (gas: 91252)
[PASS] test_proposeRetrieval_inactiveMinter() (gas: 18149)
[PASS] test_proposeRetrieval_multipleProposals() (gas: 287347)
[PASS] test_proposeRetrieval_undercollateralized() (gas: 158301)
[PASS] test_readTTGParameters() (gas: 248376)
[PASS] test_totalActiveOwedM() (gas: 40582)
[PASS] test_totalInactiveOwedM() (gas: 29461)
[PASS] test_totalOwedM() (gas: 64547)
[PASS] test_updateCollateral() (gas: 148140)
[PASS] test_updateCollateral_accrueBothPenalties() (gas: 230748)
[PASS] test_updateCollateral_futureTimestamp() (gas: 41189)
[PASS] test_updateCollateral_imposePenaltyForExpiredCollateralValue() (gas: 219929)
[PASS] test_updateCollateral_imposePenaltyForMissedCollateralUpdates() (gas: 253337)
[PASS] test_updateCollateral_inactiveMinter() (gas: 21163)
[PASS] test_updateCollateral_invalidSignatureOrder() (gas: 76845)
[PASS] test_updateCollateral_notEnoughValidSignatures() (gas: 162779)
[PASS] test_updateCollateral_shortSignature() (gas: 147970)
[PASS] test_updateCollateral_signatureArrayLengthsMismatch() (gas: 40204)
[PASS] test_updateCollateral_someSignaturesAreInvalid() (gas: 155420)
[PASS] test_updateCollateral_staleCollateralUpdate() (gas: 155776)
[PASS] test_updateCollateral_zeroThreshold() (gas: 96275)
Test result: ok. 74 passed; 0 failed; 0 skipped; finished in 24.79ms

Running 38 tests for test/MToken.t.sol:MTokenTests
[PASS] test_allowEarningOnBehalf() (gas: 39828)
[PASS] test_balanceOf_earner() (gas: 62533)
[PASS] test_balanceOf_nonEarner() (gas: 53186)
[PASS] test_burn_fromEarner() (gas: 106031)
[PASS] test_burn_fromNonEarner() (gas: 72747)
[PASS] test_burn_insufficientBalance_fromEarner() (gas: 51193)
[PASS] test_burn_insufficientBalance_fromNonEarner() (gas: 41873)
[PASS] test_burn_notMinterGateway() (gas: 11522)
[PASS] test_constructor() (gas: 11440)
[PASS] test_constructor_zeroMinterGateway() (gas: 113955)
[PASS] test_constructor_zeroTTGRegistrar() (gas: 113872)
[PASS] test_disallowEarningOnBehalf() (gas: 42055)
[PASS] test_earnerRate() (gas: 39826)
[PASS] test_emptyRateModel() (gas: 17062)
[PASS] test_latestEarnerRate() (gas: 16483)
[PASS] test_mint_notMinterGateway() (gas: 11546)
[PASS] test_mint_toEarner() (gas: 147639)
[PASS] test_mint_toNonEarner() (gas: 79513)
[PASS] test_startEarning() (gas: 134117)
[PASS] test_startEarning_notApprovedEarner() (gas: 21395)
[PASS] test_startEarning_onBehalfOf() (gas: 137473)
[PASS] test_startEarning_onBehalfOf_hasNotAllowedEarningOnBehalf() (gas: 41934)
[PASS] test_startEarning_onBehalfOf_notApprovedEarner() (gas: 46603)
[PASS] test_stopEarning() (gas: 108032)
[PASS] test_stopEarning_onBehalfOf() (gas: 109627)
[PASS] test_stopEarning_onBehalfOf_isApprovedEarner() (gas: 45246)
[PASS] test_totalEarningSupply() (gas: 54116)
[PASS] test_totalNonEarningSupply() (gas: 46614)
[PASS] test_totalSupply() (gas: 78329)
[PASS] test_totalSupply_noTotalEarningSupply() (gas: 55167)
[PASS] test_totalSupply_onlyTotalEarningSupply() (gas: 56890)
[PASS] test_transfer_fromEarner_toEarner() (gas: 117225)
[PASS] test_transfer_fromEarner_toNonEarner() (gas: 178954)
```

```
[PASS] test_transfer_fromNonEarner_toEarner() (gas: 154937)
[PASS] test_transfer_fromNonEarner_toNonEarner() (gas: 108497)
[PASS] test_transfer_insufficientBalance_fromEarner_toNonEarner() (gas: 53579)
[PASS] test_transfer_insufficientBalance_fromNonEarner_toNonEarner() (gas: 44423)
[PASS] test_updateIndex() (gas: 116061)
Test result: ok. 38 passed; 0 failed; 0 skipped; finished in 28.02ms

Running 7 tests for test/integration/minter-gateway/Integration.t.sol:IntegrationTests
[PASS] test_cancelMintProposalsAndFreezeMinter() (gas: 699529)
Logs:
  deployer:  0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Expected Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  M Token address:  0x8Ad159a275AEE56fb2334DBb69036E9c7baCEe9b
  Earner Rate Model address:  0x9c52B2C4A89E2BE37972d18dA937cbAd8AA8bd50
  Minter Rate Model address:  0xfF2Bd636B9Fc89645C2D336aeaDE2E4AbaFe1eA5

[PASS] test_compliantMinter() (gas: 1188377)
Logs:
  deployer:  0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Expected Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  M Token address:  0x8Ad159a275AEE56fb2334DBb69036E9c7baCEe9b
  Earner Rate Model address:  0x9c52B2C4A89E2BE37972d18dA937cbAd8AA8bd50
  Minter Rate Model address:  0xfF2Bd636B9Fc89645C2D336aeaDE2E4AbaFe1eA5

[PASS] test_deactivateMinterAndPayTheirInactiveOwedM() (gas: 717888)
Logs:
  deployer:  0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Expected Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  M Token address:  0x8Ad159a275AEE56fb2334DBb69036E9c7baCEe9b
  Earner Rate Model address:  0x9c52B2C4A89E2BE37972d18dA937cbAd8AA8bd50
  Minter Rate Model address:  0xfF2Bd636B9Fc89645C2D336aeaDE2E4AbaFe1eA5

[PASS] test_deactivateMinterWithMajorityOfActiveOwedM() (gas: 836940)
Logs:
  deployer:  0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Expected Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  M Token address:  0x8Ad159a275AEE56fb2334DBb69036E9c7baCEe9b
  Earner Rate Model address:  0x9c52B2C4A89E2BE37972d18dA937cbAd8AA8bd50
  Minter Rate Model address:  0xfF2Bd636B9Fc89645C2D336aeaDE2E4AbaFe1eA5

[PASS] test_earnerRateIsHigherThanMinterRate() (gas: 709795)
Logs:
  deployer:  0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Expected Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  M Token address:  0x8Ad159a275AEE56fb2334DBb69036E9c7baCEe9b
  Earner Rate Model address:  0x9c52B2C4A89E2BE37972d18dA937cbAd8AA8bd50
  Minter Rate Model address:  0xfF2Bd636B9Fc89645C2D336aeaDE2E4AbaFe1eA5

[PASS] test_nonCompliantMintersPayPenalties() (gas: 1421904)
Logs:
  deployer:  0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Expected Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  M Token address:  0x8Ad159a275AEE56fb2334DBb69036E9c7baCEe9b
  Earner Rate Model address:  0x9c52B2C4A89E2BE37972d18dA937cbAd8AA8bd50
  Minter Rate Model address:  0xfF2Bd636B9Fc89645C2D336aeaDE2E4AbaFe1eA5

[PASS] test_retrieveCollateral() (gas: 603481)
Logs:
  deployer:  0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Expected Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  M Token address:  0x8Ad159a275AEE56fb2334DBb69036E9c7baCEe9b
  Earner Rate Model address:  0x9c52B2C4A89E2BE37972d18dA937cbAd8AA8bd50
  Minter Rate Model address:  0xfF2Bd636B9Fc89645C2D336aeaDE2E4AbaFe1eA5

Test result: ok. 7 passed; 0 failed; 0 skipped; finished in 27.30ms
```

```
Running 1 test for test/invariant/Invariant.t.sol:InvariantTests
[PASS] invariant_main() (runs: 512, calls: 12800, reverts: 5327)
Logs:
  deployer:  0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Expected Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  M Token address:  0x8Ad159a275AEE56fb2334DBb69036E9c7baCEe9b
  Earner Rate Model address:  0x9c52B2C4A89E2BE37972d18dA937cbAd8AA8bd50
  Minter Rate Model address:  0xfF2Bd636B9Fc89645C2D336aeaDE2E4AbaFe1eA5
  Updating minter rate = 2453 at 1690353326
  Deactivating minter 0xB55178A219B50D6a00018b89f8ED2a12EB8322b6 with active owed M 0 at 1691217325
  Updating minter rate = 1347 at 1691457277
  Updating earner rate = 1208 at 1692321275
  Updating minter rate = 26266 at 1692334984
  Updating minter rate = 2984 at 1692353736
  Updating earner rate = 39998 at 1693217736
  Updating Minter Gateway index at 1693841053
  Updating minter rate = 103 at 1694600023
  Updating minter rate = 39925 at 1694600145
  Minting 999999999017268 M to minter 0xb7295Ffdf1bD13AF3493c353d07098a409d1deb0 at 1694945984
  Updating M Token index at 1694845451
  Minting 1000002 M to minter 0xb7295Ffdf1bD13AF3493c353d07098a409d1deb0 at 1695709451
  Updating earner rate = 100 at 1694845571
  Minting 72670449269183 M to minter 0xB59B481bf23261cE952A82060f690202d1A0528C at 1695251496
  Deactivating minter 0xB59B481bf23261cE952A82060f690202d1A0528C with active owed M 0 at 1694845694
  Updating minter rate = 9813 at 1695226509
  Updating Minter Gateway index at 1695746257
  Updating earner rate = 30679 at 1696086870
  Updating M Token index at 1696644270

Test result: ok. 1 passed; 0 failed; 0 skipped; finished in 5.56s

Running 16 tests for test/ContinuousIndexingMath.t.sol:ContinuousIndexingMathTests
[PASS] test_convertFromBasisPoints() (gas: 8064)
[PASS] test_convertToBasisPoints() (gas: 8195)
[PASS] test_divideDown() (gas: 63751)
[PASS] test_divideUp() (gas: 65573)
[PASS] test_exponent() (gas: 29571)
[PASS] test_exponentAssembly() (gas: 4960)
[PASS] test_exponentLimits() (gas: 11876)
[PASS] test_getContinuousIndex() (gas: 14567)
[PASS] test_indexLimits_dailyAt100APY() (gas: 138041240)
[PASS] test_indexLimits_dailyAt10APY() (gas: 2139126730)
[PASS] test_indexLimits_hourlyAt1000APY() (gas: 359619101)
[PASS] test_multiplyContinuousRates() (gas: 27355)
[PASS] test_multiplyDown() (gas: 67130)
[PASS] test_multiplyThenDivide_100apy() (gas: 25926)
[PASS] test_multiplyThenDivide_6apy() (gas: 26797)
[PASS] test_multiplyUp() (gas: 66408)
Test result: ok. 16 passed; 0 failed; 0 skipped; finished in 6.32s

Running 2 tests for test/fuzz/Fuzz.t.sol:FuzzTests
[PASS] testFuzz_deactivateMinter_earnerRateGreaterThanMinterRate(uint256,uint256,uint256,uint256,uint256)
(runs: 5000, μ: 798556, ~: 799122)
Logs:
  deployer:  0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Expected Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  M Token address:  0x8Ad159a275AEE56fb2334DBb69036E9c7baCEe9b
  Earner Rate Model address:  0x9c52B2C4A89E2BE37972d18dA937cbAd8AA8bd50
  Minter Rate Model address:  0xfF2Bd636B9Fc89645C2D336aeaDE2E4AbaFe1eA5

[PASS] testFuzz_earnerRateGreaterThanMinterRate(uint256,uint256,uint256,uint256,uint256) (runs: 5000, μ:
713231, ~: 713802)
Logs:
  deployer:  0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Expected Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  Minter Gateway address:  0x1240FA2A84dd9157a0e76B5Cfe98B1d52268B264
  M Token address:  0x8Ad159a275AEE56fb2334DBb69036E9c7baCEe9b
  Earner Rate Model address:  0x9c52B2C4A89E2BE37972d18dA937cbAd8AA8bd50
  Minter Rate Model address:  0xfF2Bd636B9Fc89645C2D336aeaDE2E4AbaFe1eA5
```

```
Test result: ok. 2 passed; 0 failed; 0 skipped; finished in 8.84s

Ran 11 test suites: 144 tests passed, 0 failed, 1 skipped (145 total tests)

MZero-Labs/spog
Running 9 tests for test/PureEpochs.t.sol:PureEpochsTests
[PASS] test_currentEpoch() (gas: 10357)
[PASS] test_getTimeSinceEpochEnd() (gas: 17378)
[PASS] test_getTimeSinceEpochStart() (gas: 14496)
[PASS] test_getTimeUntilEpochEnds() (gas: 27816)
[PASS] test_getTimeUntilEpochStart() (gas: 18424)
[PASS] test_getTimestampOfEpochEnd() (gas: 4315)
[PASS] test_getTimestampOfEpochStart() (gas: 3221)
[PASS] test_timeElapsedInCurrentEpoch() (gas: 15223)
[PASS] test_timeRemainingInCurrentEpoch() (gas: 16699)
Test result: ok. 9 passed; 0 failed; 0 skipped; finished in 5.17ms

Running 3 tests for test/StandardGovernorDeployer.t.sol:StandardGovernorDeployerTests
[PASS] test_deployAddress() (gas: 4812683)
[PASS] test_deployAddress_notZeroGovernor() (gas: 17747)
[PASS] test_initialState() (gas: 21012)
Test result: ok. 3 passed; 0 failed; 0 skipped; finished in 6.45ms

Running 3 tests for test/EmergencyGovernorDeployer.t.sol:EmergencyGovernorDeployerTests
[PASS] test_deployAddress() (gas: 4128819)
[PASS] test_deployAddress_notZeroGovernor() (gas: 14563)
[PASS] test_initialState() (gas: 14509)
Test result: ok. 3 passed; 0 failed; 0 skipped; finished in 6.20ms

Running 19 tests for test/Registrar.t.sol:RegistrarTests
[PASS] test_addToList_fromEmergencyGovernor() (gas: 57222)
[PASS] test_addToList_fromStandardGovernor() (gas: 51607)
[PASS] test_addToList_multiple() (gas: 121925)
[PASS] test_addToList_notStandardOrEmergencyGovernor() (gas: 22581)
[PASS] test_constructor_invalidEmergencyGovernorDeployerAddress() (gas: 44511)
[PASS] test_constructor_invalidPowerTokenDeployerAddress() (gas: 47584)
[PASS] test_constructor_invalidStandardGovernorDeployerAddress() (gas: 50735)
[PASS] test_constructor_invalidVaultAddress() (gas: 61629)
[PASS] test_constructor_invalidVoteTokenAddress() (gas: 53921)
[PASS] test_constructor_invalidZeroGovernorAddress() (gas: 37245)
[PASS] test_initialState() (gas: 51251)
[PASS] test_removeFromList_fromEmergencyGovernor() (gas: 47246)
[PASS] test_removeFromList_fromStandardGovernor() (gas: 41881)
[PASS] test_removeFromList_multiple() (gas: 106852)
[PASS] test_removeFromList_notStandardOrEmergencyGovernor() (gas: 22604)
[PASS] test_setKey_fromEmergencyGovernor() (gas: 52637)
[PASS] test_setKey_fromStandardGovernor() (gas: 47044)
[PASS] test_setKey_multiple() (gas: 117087)
[PASS] test_setKey_notStandardOrEmergencyGovernor() (gas: 20313)
Test result: ok. 19 passed; 0 failed; 0 skipped; finished in 3.41ms

Running 16 tests for test/EmergencyGovernor.t.sol:EmergencyGovernorTests
[PASS] test_addToList_callRegistrar() (gas: 19314)
[PASS] test_addToList_notSelf() (gas: 9364)
[PASS] test_constructor_invalidRegistrarAddress() (gas: 99148)
[PASS] test_constructor_invalidStandardGovernorAddress() (gas: 99230)
[PASS] test_constructor_invalidZeroGovernorAddress() (gas: 99062)
[PASS] test_initialState() (gas: 23484)
[PASS] test_removeFromAndAddToList_callRegistrar() (gas: 22931)
[PASS] test_removeFromAndAddToList_notSelf() (gas: 9647)
[PASS] test_removeFromList_callRegistrar() (gas: 19226)
[PASS] test_removeFromList_notSelf() (gas: 9230)
[PASS] test_revertIfInvalidCalldata() (gas: 10804)
[PASS] test_setKey_notSelf() (gas: 9208)
[PASS] test_setStandardProposalFee_callStandardGovernor() (gas: 16641)
[PASS] test_setStandardProposalFee_notSelf() (gas: 9093)
[PASS] test_setThresholdRatio() (gas: 19229)
[PASS] test_setThresholdRatio_notZeroGovernor() (gas: 11170)
Test result: ok. 16 passed; 0 failed; 0 skipped; finished in 10.63ms

Running 20 tests for test/ZeroGovernor.t.sol:ZeroGovernorTests
```

```
[PASS] test_constructor_invalidCashTokenAddress() (gas: 107748)
[PASS] test_constructor_invalidEmergencyGovernorDeployerAddress() (gas: 112435)
[PASS] test_constructor_invalidPowerTokenDeployerAddress() (gas: 112494)
[PASS] test_constructor_invalidStandardGovernorDeployerAddress() (gas: 112556)
[PASS] test_constructor_noAllowedCashTokens() (gas: 106934)
[PASS] test_getProposal_proposalDoesNotExist() (gas: 13274)
[PASS] test_initialState() (gas: 51426)
[PASS] test_resetToPowerHolders() (gas: 89656)
[PASS] test_resetToPowerHolders_notZeroGovernor() (gas: 8489)
[PASS] test_resetToZeroHolders() (gas: 85551)
[PASS] test_resetToZeroHolders_notZeroGovernor() (gas: 8531)
[PASS] test_revertIfInvalidCalldata() (gas: 10871)
[PASS] test_setCashToken_callStandardGovernor() (gas: 27453)
[PASS] test_setCashToken_invalidCashToken() (gas: 12438)
[PASS] test_setCashToken_notZeroGovernor() (gas: 11581)
[PASS] test_setEmergencyProposalThresholdRatio() (gas: 27044)
[PASS] test_setEmergencyProposalThresholdRatio_notZeroGovernor() (gas: 11117)
[PASS] test_setZeroProposalThresholdRatio_invalidThresholdRatioAboveOne() (gas: 11916)
[PASS] test_setZeroProposalThresholdRatio_invalidThresholdRatioBelowMin() (gas: 11895)
[PASS] test_setZeroProposalThresholdRatio_notZeroGovernor() (gas: 11120)
Test result: ok. 20 passed; 0 failed; 0 skipped; finished in 3.37ms

Running 12 tests for test/EpochBasedInflationaryVoteToken.t.sol:EpochBasedInflationaryVoteTokenTests
[PASS] test_UsersVoteInflationForMultipleEpochsWithRedelegation() (gas: 1196599)
[PASS] test_UsersVoteInflationForMultipleEpochsWithTransfers() (gas: 1456072)
[PASS] test_UsersVoteInflationUpgradeOnDelegation() (gas: 926479)
[PASS] test_UsersVoteInflationWorksWithTransfer() (gas: 783424)
[PASS] test_VotingPowerForDelegates() (gas: 1257544)
[PASS] test_inflationFromVotingPowerInPreviousEpoch_delegated() (gas: 866724)
[PASS] test_inflationFromVotingPowerInPreviousEpoch_selfDelegation() (gas: 693704)
[PASS] test_noDelegationsDuringVotingEpoch() (gas: 17859)
[PASS] test_noInflationWithoutVotingPowerInPreviousEpoch_delegated() (gas: 714770)
[PASS] test_noInflationWithoutVotingPowerInPreviousEpoch_selfDelegation() (gas: 613124)
[PASS] test_noTransfersDuringVotingEpoch() (gas: 217691)
[PASS] test_scenario1() (gas: 1628627)
Test result: ok. 12 passed; 0 failed; 0 skipped; finished in 12.55ms

Running 3 tests for test/integration/Integration.t.sol:IntegrationTests
[PASS] test_emergencySetKey() (gas: 235148)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

[PASS] test_initialState() (gas: 132288)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

[PASS] test_setKey() (gas: 741621)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

Test result: ok. 3 passed; 0 failed; 0 skipped; finished in 15.30ms
```

```
Running 1 test for test/integration/auction/auction.t.sol:Auction_IntegrationTest
[PASS] test_auction_multipleEpochs() (gas: 1857722)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

Test result: ok. 1 passed; 0 failed; 0 skipped; finished in 17.78ms

Running 3 tests for test/PowerTokenDeployer.t.sol:DeployerTests
[PASS] test_deployAddress() (gas: 5457369)
[PASS] test_deployAddress_notZeroGovernor() (gas: 16744)
[PASS] test_initialState() (gas: 14509)
Test result: ok. 3 passed; 0 failed; 0 skipped; finished in 1.20ms

Running 4 tests for test/integration/inflation-
rewards/powerInflationZeroRewards.t.sol:PowerInflationZeroRewards_IntegrationTest
[PASS] test_powerInflation_multiplDelegatesTransfersAndRedelegations() (gas: 2201565)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

[PASS] test_powerInflation_selfDelegationOnlyNoTransfersOrRedelegations() (gas: 2572392)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

[PASS] test_powerInflation_selfDelegationOnlyTransfersAndRedelegations() (gas: 2479180)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

[PASS] test_zeroRewards_multiplDelegatesTransfersAndRedelegations() (gas: 2013399)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

Test result: ok. 4 passed; 0 failed; 0 skipped; finished in 21.73ms

Running 1 test for test/integration/zero-governor/set-cash-
token/setCashToken.t.sol:SetCashToken_IntegrationTest
[PASS] test_zeroGovernorProposal_setCashToken() (gas: 257539)
```

```
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

Test result: ok. 1 passed; 0 failed; 0 skipped; finished in 6.52ms

Running 16 tests for test/PowerToken.t.sol:PowerTokenTests
[PASS] test_amountToAuction() (gas: 316165)
[PASS] test_buy() (gas: 258727)
[PASS] test_buy_insufficientAuctionSupply() (gas: 14883)
[PASS] test_buy_notInVotePeriod() (gas: 31683)
[PASS] test_buy_transferFromFailed() (gas: 76792)
[PASS] test_constructor_invalidBootstrapTokenAddress() (gas: 120970)
[PASS] test_constructor_invalidCashTokenAddress() (gas: 123284)
[PASS] test_constructor_invalidStandardGovernorAddress() (gas: 121032)
[PASS] test_constructor_invalidVaultAddress() (gas: 143269)
[PASS] test_getCost() (gas: 329411)
[PASS] test_initialState() (gas: 147717)
[PASS] test_notAffectedByBootstrapTokenAfterBootstrapEpoch() (gas: 13412420)
[PASS] test_setNextCashToken_NotStandardGovernor() (gas: 9043)
[PASS] test_setNextCashToken_afterNextCashTokenStartingEpoch() (gas: 57604)
[PASS] test_setNextCashToken_beforeNextCashTokenStartingEpoch() (gas: 63895)
[PASS] test_setNextCashToken_invalidCashTokenAddress() (gas: 11809)
Test result: ok. 16 passed; 0 failed; 0 skipped; finished in 13.70ms

Running 25 tests for test/ZeroToken.t.sol:ZeroTokenTests
[PASS] test_getPastVotes_multi_afterAllSnaps() (gas: 139670)
[PASS] test_getPastVotes_multi_beforeAllSnaps() (gas: 87989)
[PASS] test_getPastVotes_multi_notPastTimepoint() (gas: 25846)
[PASS] test_getPastVotes_multi_single() (gas: 93418)
[PASS] test_getPastVotes_multi_startEpochAfterEndEpoch() (gas: 15238)
[PASS] test_getPastVotes_multi_subset() (gas: 173726)
[PASS] test_initialState() (gas: 72090)
[PASS] test_pastBalancesOf_afterAllSnaps() (gas: 139802)
[PASS] test_pastBalancesOf_beforeAllSnaps() (gas: 88014)
[PASS] test_pastBalancesOf_notPastTimepoint() (gas: 25825)
[PASS] test_pastBalancesOf_single() (gas: 93417)
[PASS] test_pastBalancesOf_startEpochAfterEndEpoch() (gas: 15282)
[PASS] test_pastBalancesOf_subset() (gas: 173881)
[PASS] test_pastDelegates_multi_afterAllSnaps() (gas: 120072)
[PASS] test_pastDelegates_multi_beforeAllSnaps() (gas: 118598)
[PASS] test_pastDelegates_multi_notPastTimepoint() (gas: 25821)
[PASS] test_pastDelegates_multi_single() (gas: 123184)
[PASS] test_pastDelegates_multi_startEpochAfterEndEpoch() (gas: 15235)
[PASS] test_pastDelegates_multi_subset() (gas: 128883)
[PASS] test_pastTotalSupplies_afterAllSnaps() (gas: 117650)
[PASS] test_pastTotalSupplies_beforeAllSnaps() (gas: 69451)
[PASS] test_pastTotalSupplies_notPastTimepoint() (gas: 22952)
[PASS] test_pastTotalSupplies_single() (gas: 72537)
[PASS] test_pastTotalSupplies_startEpochAfterEndEpoch() (gas: 12669)
[PASS] test_pastTotalSupplies_subset() (gas: 151181)
Test result: ok. 25 passed; 0 failed; 0 skipped; finished in 16.26ms

Running 4 tests for test/DistributionVault.t.sol:DistributionVaultTests
[PASS] test_constructor() (gas: 19413)
[PASS] test_constructor_invalidZeroTokenAddress() (gas: 62550)
[PASS] test_distribution() (gas: 1093039)
[PASS] test_getClaimable_notPastTimepoint() (gas: 20533)
Test result: ok. 4 passed; 0 failed; 0 skipped; finished in 30.20ms

Running 5 tests for test/integration/standard-
governor/propose/standardGovernorPropose.t.sol:StandardGovernorPropose_IntegrationTest
[PASS] test_standardGovernorPropose_changeProposalFee() (gas: 963755)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
```

```
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

[PASS] test_standardGovernorPropose_proposalLifecycle() (gas: 2164436)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

[PASS] test_standardGovernorPropose_proposalPendingActiveDefeated() (gas: 286558)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

[PASS] test_standardGovernorPropose_proposalPendingActiveSucceededExecuted() (gas: 805531)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

[PASS] test_standardGovernorPropose_proposalPendingActiveSucceededExpired() (gas: 772527)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

Test result: ok. 5 passed; 0 failed; 0 skipped; finished in 9.57ms

Running 1 test for test/integration/zero-governor/set-
thresholds/setZeroEmergencyThresholds.t.sol:SetZeroAndEmergencyThresholds_IntegrationTest
[PASS] test_zeroGovernorProposal_setZeroAndEmergencyThresholds() (gas: 332029)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

Test result: ok. 1 passed; 0 failed; 0 skipped; finished in 9.05ms

Running 1 test for test/integration/zero-governor/reset/reset-to-zero-
holders/resetToZeroHolders.t.sol:ResetToZeroHolders_IntegrationTest
[PASS] test_resetToZeroHolders() (gas: 15511221)
Logs:
```

```
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

Test result: ok. 1 passed; 0 failed; 0 skipped; finished in 9.58ms

Running 1 test for test/integration/zero-governor/reset/reset-to-power-
holders/resetToPowerHolders.t.sol:ResetToPowerHolders_IntegrationTest
[PASS] test_resetToPowerHolders() (gas: 15686362)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

Test result: ok. 1 passed; 0 failed; 0 skipped; finished in 10.98ms

Running 5 tests for test/integration/zero-
governor/propose/zeroGovernorPropose.t.sol:ZeroGovernorPropose_IntegrationTest
[PASS] test_zeroGovernorPropose_proposalActiveDefeated() (gas: 178234)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

[PASS] test_zeroGovernorPropose_proposalActiveExpired() (gas: 104271)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

[PASS] test_zeroGovernorPropose_proposalActiveSucceededExecuted() (gas: 14464553)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

[PASS] test_zeroGovernorPropose_proposalActiveSucceededExpired() (gas: 286835)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b
```

```
[PASS] test_zeroGovernorPropose_totalSupplyZero() (gas: 114804)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

Test result: ok. 5 passed; 0 failed; 0 skipped; finished in 7.32ms

Running 4 tests for test/integration/vault/distributionVault.t.sol:DistributionVault_IntegrationTest
[PASS] test_distributeAndClaim_ZeroPowerWeightsStayTheSame() (gas: 661577)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

[PASS] test_distributeInMultipleEpochsAndClaimMultipleTimes() (gas: 809527)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

[PASS] test_distributeInMultipleEpochsAndClaimOnce_ZeroPowerWeightsChange() (gas: 668092)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

[PASS] test_distributeInMultipleEpochsAndGetClaimable_ZeroPowerWeightsChange() (gas: 963530)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

Test result: ok. 4 passed; 0 failed; 0 skipped; finished in 23.36ms

Running 4 tests for test/integration/emergency-
governor/propose/emergencyGovernorPropose.t.sol:EmergencyGovernorPropose_IntegrationTest
[PASS] test_emergencyGovernorPropose_proposalActiveDefeated() (gas: 274804)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b
```

```
[PASS] test_emergencyGovernorPropose_proposalActiveDefeatedFast() (gas: 509892)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

[PASS] test_emergencyGovernorPropose_proposalActiveSucceededExecuted() (gas: 445336)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

[PASS] test_emergencyGovernorPropose_proposalActiveSucceededExpired() (gas: 275518)
Logs:
  deployer: 0xaE0bDc4eEAC5E950B67C6819B118761CaAF61946
  Registrar Address: 0x13250CF16EEc77781DCF240b067cAC78F2b2Adf8
  Power Token Address: 0xF0C36E5Bf7a10DeBaE095410c8b1A6E9501DC0f7
  Zero Token Address: 0xa5906e11c3b7F5B832bcBf389295D44e7695b4A6
  Standard Governor Address: 0xcF9F374922476C09607b9dcFF1fCA397BABE0b0C
  Emergency Governor Address: 0x0d5C87e3905Da4B351d605a0d89953aF60eF667a
  Zero Governor Address: 0x9101223D33eEaeA94045BB2920F00BA0F7A475Bc
  Distribution Vault Address: 0x8584361C55e82129246aDAEb93E6a2b4d4C7891b

Test result: ok. 4 passed; 0 failed; 0 skipped; finished in 22.01ms

Running 12 tests for test/EpochBasedVoteToken.t.sol:EpochBasedVoteTokenTests
[PASS] test_balanceOf() (gas: 58862)
[PASS] test_delegates() (gas: 61266)
[PASS] test_getPastVotes() (gas: 102112)
[PASS] test_getPastVotes_notPastTimepoint() (gas: 21213)
[PASS] test_getVotes() (gas: 58908)
[PASS] test_pastBalanceOf() (gas: 102265)
[PASS] test_pastBalanceOf_notPastTimepoint() (gas: 21301)
[PASS] test_pastDelegate() (gas: 148679)
[PASS] test_pastDelegates_notPastTimepoint() (gas: 21353)
[PASS] test_pastTotalSupply() (gas: 95469)
[PASS] test_pastTotalSupply_notPastTimepoint() (gas: 18179)
[PASS] test_totalSupply() (gas: 55478)
Test result: ok. 12 passed; 0 failed; 0 skipped; finished in 18.04ms

Running 51 tests for test/StandardGovernor.t.sol:StandardGovernorTests
[PASS] test_addToList_callRegistrar() (gas: 19249)
[PASS] test_addToList_notSelf() (gas: 11373)
[PASS] test_castVote_alreadyVoted() (gas: 76285)
[PASS] test_castVote_notActive() (gas: 49877)
[PASS] test_castVote_voteNo() (gas: 174733)
[PASS] test_castVote_voteYes() (gas: 174767)
[PASS] test_castVote_votedOnAllProposalsMultipleProposalExists() (gas: 385392)
[PASS] test_castVote_votedOnAllProposalsOnlyOneProposalExists() (gas: 211685)
[PASS] test_castVote_votedOnFirstOfSeveralProposals() (gas: 198720)
[PASS] test_castVotes() (gas: 292896)
[PASS] test_castVotes_multipleTimes() (gas: 375270)
[PASS] test_constructor_invalidEmergencyGovernorDeployerAddress() (gas: 86995)
[PASS] test_constructor_invalidRegistrarAddress() (gas: 87096)
[PASS] test_constructor_invalidVaultAddress() (gas: 87223)
[PASS] test_constructor_invalidVoteTokenAddress() (gas: 86952)
[PASS] test_constructor_invalidZeroGovernorAddress() (gas: 87099)
[PASS] test_constructor_invalidZeroTokenAddress() (gas: 87239)
[PASS] test_execute_proposalCannotBeExecuted() (gas: 59283)
[PASS] test_initialState() (gas: 38977)
[PASS] test_propose_invalidCallData() (gas: 16636)
```

```
[PASS] test_propose_invalidCallDatasLength() (gas: 17306)
[PASS] test_propose_invalidTarget() (gas: 13656)
[PASS] test_propose_invalidTargetsLength() (gas: 14215)
[PASS] test_propose_invalidValue() (gas: 14614)
[PASS] test_propose_invalidValuesLength() (gas: 14988)
[PASS] test_propose_proposalExists() (gas: 291191)
[PASS] test_propose_proposalExists_withHarness() (gas: 59075)
[PASS] test_propose_uniqueProposalIds() (gas: 283286)
[PASS] test_quorum() (gas: 7540)
[PASS] test_removeFromAndAddToList_notSelf() (gas: 13903)
[PASS] test_removeFromList_callRegistrar() (gas: 19358)
[PASS] test_removeFromList_notSelf() (gas: 11439)
[PASS] test_revertIfInvalidCalldata() (gas: 10850)
[PASS] test_sendProposalFeeToVault() (gas: 75012)
[PASS] test_sendProposalFeeToVault_feeNotDestinedForVault() (gas: 91362)
[PASS] test_sendProposalFeeToVault_noFeeToSend() (gas: 73902)
[PASS] test_setCashToken() (gas: 42195)
[PASS] test_setCashToken_invalidCashTokenAddress() (gas: 14129)
[PASS] test_setCashToken_notZeroGovernor() (gas: 13762)
[PASS] test_setKey_notSelf() (gas: 9254)
[PASS] test_setProposalFee_byEmergencyGovernor() (gas: 18887)
[PASS] test_setProposalFee_bySelf() (gas: 16885)
[PASS] test_setProposalFee_notSelf() (gas: 9153)
[PASS] test_state_activeThenDefeatedMajorityVotedNo() (gas: 182339)
[PASS] test_state_activeThenDefeatedNobodyVoted() (gas: 46982)
[PASS] test_state_activeThenSucceededMajorityVotedYes() (gas: 182407)
[PASS] test_state_executed() (gas: 61570)
[PASS] test_state_pendingThenActive() (gas: 46845)
[PASS] test_state_succeededThenExpired() (gas: 75961)
[PASS] test_votingDelay() (gas: 17470)
[PASS] test_votingPeriod() (gas: 5843)
Test result: ok. 51 passed; 0 failed; 0 skipped; finished in 24.97ms

Running 1 test for
test/fuzz/EpochBasedInflationaryVoteTokenFuzz.t.sol:EpochBasedInflationaryVoteTokenFuzzTests
[PASS] testFuzz_full(uint256) (runs: 256, μ: 290481413, ~: 290546223)
Test result: ok. 1 passed; 0 failed; 0 skipped; finished in 146.04s

Ran 25 test suites: 224 tests passed, 0 failed, 0 skipped (224 total tests)
```

Code Coverage

We note that coverage, while good overall, does have spots where it can be improved.

MZero-Labs/common

File	% Lines	% Statements	% Branches	% Funcs
src/ContractHelper.sol	100.00% (**2/**2)	100.00% (**2/**2)	100.00% (**0/**0)	100.00% (**1/**1)
src/ERC20Extended.sol	0.00% (**0/**18)	0.00% (**0/**19)	0.00% (**0/**2)	0.00% (**0/**8)
src/ERC3009.sol	100.00% (**35/**35)	100.00% (**39/**39)	100.00% (**10/**10)	100.00% (**16/**16)
src/ERC712.sol	52.94% (**9/**17)	50.00% (**13/**26)	50.00% (**6/**12)	55.56% (**5/**9)
src/libs/SignatureChecker.sol	100.00% (**35/**35)	100.00% (**70/**70)	100.00% (**8/**8)	100.00% (**14/**14)
src/libs/UIntMath.sol	100.00% (**22/**22)	100.00% (**44/**44)	100.00% (**12/**12)	100.00% (**16/**16)
test/ContractHelper.t.sol	100.00% (**1/**1)	100.00% (**1/**1)	100.00% (**0/**0)	100.00% (**1/**1)

File	% Lines	% Statements	% Branches	% Funcs
test/SignatureChecker.t.sol	75.00% (**3/**4)	80.00% (**4/**5)	100.00% (**0/**0)	71.43% (**5/**7)
test/utils/ContractHelperHarness.sol	100.00% (**1/**1)	100.00% (**2/**2)	100.00% (**0/**0)	100.00% (**1/**1)
test/utils/ERC20ExtendedHarness.sol	80.00% (**4/**5)	77.78% (**7/**9)	100.00% (**0/**0)	62.50% (**5/**8)
test/utils/SignatureCheckerHarness.sol	100.00% (**13/**13)	100.00% (**26/**26)	100.00% (**0/**0)	100.00% (**13/**13)
test/utils/TestUtils.t.sol	0.00% (**0/**6)	0.00% (**0/**8)	0.00% (**0/**2)	0.00% (**0/**4)
test/utils/UIntMathHarness.sol	100.00% (**16/**16)	100.00% (**32/**32)	100.00% (**0/**0)	100.00% (**16/**16)
Total	80.57% (**141/**175)	84.81% (**240/**283)	78.26% (**36/**46)	81.58% (**93/**114)

MZero-Labs/protocol

File	% Lines	% Statements	% Branches	% Funcs
script/Deploy.s.sol	0.00% (**0/**2)	0.00% (**0/**3)	100.00% (**0/**0)	0.00% (**0/**1)
script/DeployBase.s.sol	93.33% (**14/**15)	94.44% (**17/**18)	0.00% (**0/**2)	100.00% (**1/**1)
src/MToken.sol	96.67% (**87/**90)	93.52% (**101/**108)	83.33% (**20/**24)	93.55% (**29/**31)
src/MinterGateway.sol	100.00% (**217/**217)	98.19% (**326/**332)	91.46% (**75/**82)	96.61% (**57/**59)
src/abstract/ContinuousIndexing.sol	100.00% (**16/**16)	100.00% (**26/**26)	100.00% (**2/**2)	100.00% (**10/**10)
src/libs/ContinuousIndexingMath.sol	87.50% (**14/**16)	87.88% (**29/**33)	50.00% (**2/**4)	90.00% (**9/**10)
src/libs/TTGRegistrarReader.sol	94.74% (**18/**19)	95.00% (**38/**40)	100.00% (**0/**0)	94.74% (**18/**19)
src/rateModels/MinterRateModel.sol	100.00% (**2/**2)	100.00% (**4/**4)	100.00% (**0/**0)	50.00% (**1/**2)
src/rateModels/SplitEarnerRateModel.sol	0.00% (**0/**7)	0.00% (**0/**13)	0.00% (**0/**4)	0.00% (**0/**2)
src/rateModels/StableEarnerRateModel.sol	100.00% (**14/**14)	92.86% (**26/**28)	80.00% (**8/**10)	66.67% (**2/**3)
test/integration/IntegrationBaseSetup.t.sol	0.00% (**0/**59)	0.00% (**0/**71)	100.00% (**0/**0)	0.00% (**0/**5)
test/invariant/Invariant.t.sol	69.70% (**69/**99)	66.96% (**77/**115)	50.00% (**2/**4)	80.00% (**12/**15)
test/utils/ContinuousIndexingMathHarness.sol	100.00% (**9/**9)	100.00% (**18/**18)	100.00% (**0/**0)	100.00% (**9/**9)

File	% Lines	% Statements	% Branches	% Funcs
test/utils/DigestHelper.sol	100.00% (**5/**5)	100.00% (**6/**6)	100.00% (**0/**0)	33.33% (**1/**3)
test/utils/MTokenHarness.sol	88.89% (**8/**9)	90.00% (**9/**10)	100.00% (**0/**0)	88.89% (**8/**9)
test/utils/MinterGatewayHarness.sol	95.45% (**21/**22)	96.15% (**25/**26)	100.00% (**0/**0)	95.45% (**21/**22)
test/utils/Mocks.sol	76.19% (**16/**21)	76.19% (**16/**21)	100.00% (**2/**2)	77.27% (**17/**22)
test/utils/TestUtils.sol	0.00% (**0/**15)	0.00% (**0/**20)	0.00% (**0/**2)	0.00% (**0/**7)
Total	80.06% (**510/**637)	80.49% (**718/**892)	81.62% (**111/**136)	84.78% (**195/**230)

MZero-Labs/spog

File	% Lines	% Statements	% Branches	% Funcs
script/Deploy.s.sol	0.00% (**0/**2)	0.00% (**0/**3)	100.00% (**0/**0)	0.00% (**0/**1)
script/DeployBase.s.sol	97.62% (**41/**42)	98.00% (**49/**50)	0.00% (**0/**2)	100.00% (**9/**9)
src/DistributionVault.sol	100.00% (**34/**34)	98.04% (**50/**51)	83.33% (**5/**6)	100.00% (**9/**9)
src/EmergencyGovernor.sol	100.00% (**16/**16)	100.00% (**21/**21)	100.00% (**2/**2)	100.00% (**9/**9)
src/EmergencyGovernorDeployer.sol	100.00% (**4/**4)	100.00% (**5/**5)	100.00% (**0/**0)	100.00% (**2/**2)
src/PowerBootstrapToken.sol	100.00% (**2/**2)	100.00% (**2/**2)	100.00% (**0/**0)	100.00% (**2/**2)
src/PowerToken.sol	100.00% (**76/**76)	96.67% (**116/**120)	91.18% (**31/**34)	100.00% (**22/**22)
src/PowerTokenDeployer.sol	100.00% (**3/**3)	100.00% (**5/**5)	100.00% (**0/**0)	100.00% (**2/**2)
src/Registrar.sol	100.00% (**20/**20)	100.00% (**37/**37)	100.00% (**4/**4)	100.00% (**13/**13)
src/StandardGovernor.sol	96.34% (**79/**82)	93.04% (**107/**115)	81.25% (**26/**32)	100.00% (**25/**25)
src/StandardGovernorDeployer.sol	100.00% (**4/**4)	100.00% (**5/**5)	100.00% (**0/**0)	100.00% (**2/**2)
src/ZeroGovernor.sol	96.43% (**27/**28)	95.00% (**38/**40)	75.00% (**6/**8)	100.00% (**11/**11)
src/ZeroToken.sol	100.00% (**43/**43)	100.00% (**62/**62)	100.00% (**8/**8)	87.50% (**7/**8)
src/abstract/BatchGovernor.sol	77.46% (**55/**71)	78.63% (**92/**117)	90.62% (**29/**32)	62.86% (**22/**35)
src/abstract/ERC5805.sol	10.00% (**1/**10)	7.14% (**1/**14)	0.00% (**0/**2)	25.00% (**1/**4)
src/abstract/EpochBasedInflationaryVoteToken.sol	98.04% (**50/**51)	93.75% (**75/**80)	80.00% (**16/**20)	100.00% (**17/**17)

File	% Lines	% Statements	% Branches	% Funcs
src/abstract/EpochBasedVoteToken.sol	92.93% (**92/**99)	93.66% (**133/**142)	100.00% (**16/**16)	89.74% (**35/**39)
src/abstract/ThresholdGovernor.sol	91.89% (**34/**37)	87.27% (**48/**55)	87.50% (**14/**16)	80.00% (**8/**10)
src/libs/PureEpochs.sol	44.44% (**4/**9)	35.48% (**11/**31)	100.00% (**0/**0)	44.44% (**4/**9)
test/integration/IntegrationBaseSetup.t.sol	0.00% (**0/**16)	0.00% (**0/**19)	100.00% (**0/**0)	0.00% (**0/**1)
test/integration/zero-**governor/reset/**ResetIntegrationBaseSetup.t.sol	0.00% (**0/**33)	0.00% (**0/**43)	100.00% (**0/**0)	0.00% (**0/**1)
test/utils/ERC20ExtendedHarness.sol	87.50% (**7/**8)	87.50% (**7/**8)	100.00% (**0/**0)	75.00% (**3/**4)
test/utils/EmergencyGovernorHarness.sol	100.00% (**1/**1)	100.00% (**1/**1)	100.00% (**0/**0)	100.00% (**1/**1)
test/utils/EpochBasedInflationaryVoteTokenHarness.sol	100.00% (**2/**2)	100.00% (**2/**2)	100.00% (**0/**0)	100.00% (**2/**2)
test/utils/EpochBasedVoteTokenHarness.sol	80.00% (**4/**5)	80.00% (**4/**5)	100.00% (**0/**0)	80.00% (**4/**5)
test/utils/Invariants.sol	0.00% (**0/**14)	0.00% (**0/**20)	100.00% (**0/**0)	0.00% (**0/**2)
test/utils/Mocks.sol	95.12% (**39/**41)	96.00% (**48/**50)	100.00% (**0/**0)	95.45% (**42/**44)
test/utils/PowerTokenHarness.sol	77.78% (**7/**9)	77.78% (**7/**9)	100.00% (**0/**0)	77.78% (**7/**9)
test/utils/StandardGovernorHarness.sol	85.71% (**6/**7)	87.50% (**7/**8)	100.00% (**0/**0)	85.71% (**6/**7)
test/utils/TestUtils.sol	0.00% (**0/**20)	0.00% (**0/**31)	100.00% (**0/**0)	0.00% (**0/**16)
test/utils/ZeroGovernorHarness.sol	100.00% (**1/**1)	100.00% (**1/**1)	100.00% (**0/**0)	100.00% (**1/**1)
test/utils/ZeroTokenHarness.sol	100.00% (**4/**4)	100.00% (**4/**4)	100.00% (**0/**0)	100.00% (**4/**4)
Total	82.62% (**656/**794)	81.14% (**938/**1156)	86.26% (**157/**182)	82.82% (**270/**326)

Changelog

- 2024-02-01 - Initial report
- 2024-03-07 - Final report

About Quantstamp

Quantstamp is a global leader in blockchain security. Founded in 2017, Quantstamp's mission is to securely onboard the next billion users to Web3 through its best-in-class Web3 security products and services.

Quantstamp's team consists of cybersecurity experts hailing from globally recognized organizations including Microsoft, AWS, BMW, Meta, and the Ethereum Foundation. Quantstamp engineers hold PhDs or advanced computer science degrees, with decades of combined experience in formal verification, static analysis, blockchain audits, penetration testing, and original leading-edge research.

To date, Quantstamp has performed more than 500 audits and secured over $200 billion in digital asset risk from hackers. Quantstamp has worked with a diverse range of customers, including startups, category leaders and financial institutions. Brands that Quantstamp has worked with include Ethereum 2.0, Binance, Visa, PayPal, Polygon, Avalanche, Curve, Solana, Compound, Lido, MakerDAO, Arbitrum, OpenSea and the World Economic Forum.

Quantstamp's collaborations and partnerships showcase our commitment to world-class research, development and security. We're honored to work with some of the top names in the industry and proud to secure the future of web3.

Notable Collaborations & Customers:

- Blockchains: Ethereum 2.0, Near, Flow, Avalanche, Solana, Cardano, Binance Smart Chain, Hedera Hashgraph, Tezos
- DeFi: Curve, Compound, Maker, Lido, Polygon, Arbitrum, SushiSwap
- NFT: OpenSea, Parallel, Dapper Labs, Decentraland, Sandbox, Axie Infinity, Illuvium, NBA Top Shot, Zora
- Academic institutions: National University of Singapore, MIT

Timeliness of content

The content contained in the report is current as of the date appearing on the report and is subject to change without notice, unless indicated otherwise by Quantstamp; however, Quantstamp does not guarantee or warrant the accuracy, timeliness, or completeness of any report you access using the internet or other means, and assumes no obligation to update any information following publication or other making available of the report to you by Quantstamp.

Notice of confidentiality

This report, including the content, data, and underlying methodologies, are subject to the confidentiality and feedback provisions in your agreement with Quantstamp. These materials are not to be disclosed, extracted, copied, or distributed except to the extent expressly authorized by Quantstamp.

Links to other websites

You may, through hypertext or other computer links, gain access to web sites operated by persons other than Quantstamp. Such hyperlinks are provided for your reference and convenience only, and are the exclusive responsibility of such web sites&aspo; owners. You agree that Quantstamp are not responsible for the content or operation of such web sites, and that Quantstamp shall have no liability to you or any other person or entity for the use of third-party web sites. Except as described below, a hyperlink from this web site to another web site does not imply or mean that Quantstamp endorses the content on that web site or the operator or operations of that site. You are solely responsible for determining the extent to which you may use any content at any other web sites to which you link from the report. Quantstamp assumes no responsibility for the use of third-party software on any website and shall have no liability whatsoever to any person or entity for the accuracy or completeness of any output generated by such software.

Disclaimer

The review and this report are provided on an as-is, where-is, and as-available basis. To the fullest extent permitted by law, Quantstamp disclaims all warranties, expressed or implied, in connection with this report, its content, and the related services and products and your use thereof, including, without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement. You agree that your access and/or use of the report and other results of the review, including but not limited to any associated services, products, protocols, platforms, content, and materials, will be at your sole risk. FOR AVOIDANCE OF DOUBT, THE REPORT, ITS CONTENT, ACCESS, AND/OR USAGE THEREOF, INCLUDING ANY ASSOCIATED SERVICES OR MATERIALS, SHALL NOT BE CONSIDERED OR RELIED UPON AS ANY FORM OF FINANCIAL, INVESTMENT, TAX, LEGAL, REGULATORY, OR OTHER ADVICE. This report is based on the scope of materials and documentation provided for a limited review at the time provided. You acknowledge that Blockchain technology remains under development and is subject to unknown risks and flaws and, as such, the report may not be complete or inclusive of all vulnerabilities. The review is limited to the materials identified in the report and does not extend to the compiler layer, or any other areas beyond the programming language, or other programming aspects that could present security risks. The report does not indicate the endorsement by Quantstamp of any particular project or team, nor guarantee its security, and may not be represented as such. No third party is entitled to rely on the report in any way, including for the purpose of making any decisions to buy or sell a product, service or any other asset. Quantstamp does not warrant, endorse, guarantee, or assume responsibility for any product or service advertised or offered by a third party, or any open source or third-party software, code, libraries, materials, or information linked to, called by, referenced by or accessible through the report, its content, or any related services and products, any hyperlinked websites, or any other websites or mobile applications, and we will not be a party to or in any way be responsible for monitoring any transaction between you and any third party. As with the purchase or use of a product or service through any medium or in any environment, you should use your best judgment and exercise caution where appropriate.