StakeWise

StakeWise v3

Smart Contract Security Review

Version: 2.1

June, 2024

Contents

Introduction

Sigma Prime was commercially engaged to perform a time-boxed security review of the StakeWise smart contracts. The review focused solely on the security aspects of the Solidity implementation of the contract, though general recommendations and informational comments are also provided.

Disclaimer

Sigma Prime makes all effort but holds no responsibility for the findings of this security review. Sigma Prime does not provide any guarantees relating to the function of the smart contract. Sigma Prime makes no judgements on, or provides any security review, regarding the underlying business model or the individuals involved in the project.

Document Structure

The first section provides an overview of the functionality of the StakeWise smart contracts contained within the scope of the security review. A summary followed by a detailed review of the discovered vulnerabilities is then given which assigns each vulnerability a severity rating (see Vulnerability Severity Classification), an *open/closed/resolved* status and a recommendation. Additionally, findings which do not have direct security implications (but are potentially of interest) are marked as *informational*.

Outputs of automated testing that were developed during this assessment are also included for reference (in the Appendix: Test Suite).

The appendix provides additional documentation, including the severity matrix used to classify vulnerabilities within the StakeWise smart contracts.

Overview

StakeWise is a non-custodial, decentralised staking solution that launched in early 2021. StakeWise recently underwent a major upgrade, introducing StakeWise V3, a brand new model for liquid staking allowing anyone to stake on their own terms.

StakeWise V3 acts as a white-labelled Liquid Staking Solution, allowing any node operator or DApp to launch its own liquid staking solution by leveraging the V3 architecture.

StakeWise infrastructure, combined with tailored tokenomics, aims to provide high staking yields for its users. As a liquid staking platform, users are free to un-stake at any time or utilise their staked ETH capital to earn enhanced yields throughout DeFi.

Security Assessment Summary

Scope

The scope of this time-boxed review was strictly limited to files at commit 919b273.

The fixes to the raised issues were assessed at the commit diff 919b273..0bd3c39 of PR #91.

Note: third party libraries and dependencies, such as OpenZeppelin, were excluded from the scope of this assessment.

Approach

The manual review focused on identifying issues associated with the business logic implementation of the contracts. This includes their internal interactions, intended functionality and correct implementation with respect to the underlying functionality of the Ethereum Virtual Machine (for example, verifying correct storage/memory layout).

Additionally, the manual review process focused on identifying vulnerabilities related to known Solidity anti-patterns and attack vectors, such as re-entrancy, front-running, integer overflow/underflow and correct visibility specifiers.

For a more detailed, but non-exhaustive list of examined vectors, see [1, 2].

To support this review, the testing team also utilised the following automated testing tools:

- Mythril: `https://github.com/ConsenSys/mythril`
- Slither: `https://github.com/trailofbits/slither`
- Surya: `https://github.com/ConsenSys/surya`

Output for these automated tools is available upon request.

Coverage Limitations

Due to a time-boxed nature of this review, all documented vulnerabilities reflect best effort within the allotted, limited engagement time. As such, Sigma Prime recommends to further investigate areas of the code, and any related functionality, where majority of critical and high risk vulnerabilities were identified.

Findings Summary

The testing team identified a total of 12 issues during this assessment. Categorised by their severity:

- Critical: 1 issue.
- Medium: 2 issues.
- Low: 5 issues.

- Informational: 4 issues.

Detailed Findings

This section provides a detailed description of the vulnerabilities identified within the StakeWise smart contracts. Each vulnerability has a severity classification which is determined from the likelihood and impact of each issue by the matrix given in the Appendix: Vulnerability Severity Classification.

A number of additional properties of the contracts, including gas optimisations, are also described in this section and are labelled as “informational”.

Each vulnerability is also assigned a **status**:

- ***Open:*** the issue has not been addressed by the project team.
- ***Resolved:*** the issue was acknowledged by the project team and updates to the affected contract(s) have been made to mitigate the related risk.
- ***Closed:*** the issue was acknowledged by the project team but no further actions have been taken.

Summary of Findings

ID	Description	Severity	Status
STW2-01	Delayed Fee Accounting Enables Reward Theft	Critical	Closed
STW2-02	Uncallable `upgradeToAndCall()`	Medium	Resolved
STW2-03	Potential OsToken Supply Inflation	Medium	Closed
STW2-04	Users Can Block Ejection From Fox Vault	Low	Resolved
STW2-05	Price Feed Can Depeg From Real xDai Value	Low	Closed
STW2-06	Temporary Incorrect Reward On Zero `totalShares`	Low	Closed
STW2-07	Data Type Inconsistency May Cause Precision Loss	Low	Resolved
STW2-08	Potential Arithmetic Underflow	Low	Closed
STW2-09	Inconsistency Upon Value Update	Informational	Closed
STW2-10	Potential Incorrect Use Of `MerkleProof.multiProofVerifyCalldata()`	Informational	Closed
STW2-11	Potential Off-By-One Error	Informational	Closed
STW2-12	Miscellaneous General Comments	Informational	Closed

STW2-01	Delayed Fee Accounting Enables Reward Theft		
Asset	`vaults/gnosis/*`		
Status	**Closed:** See Resolution		
Rating	Severity: Critical	Impact: High	Likelihood: High

Description

Due to how rewards are collected in Gnosis based vaults it is possible for accumulated rewards to be stolen by a new depositor.

When receiving staking and Mev rewards in the `xDai` token, the system does not update user asset allocations immediately. This is because conversion from `xDai` to `GNO` is required to be completed first. This conversion occurs via `VaultGnoStaking.swapXdaiToGno()`, which can be called by anyone, and once called will update the assets allocated to all current depositors.

This means a malicious user can notice a large balance of `xDai` pending conversion to `GNO` and enter a vault as a depositor just prior to this conversion and steal rewards relating to the period prior to their entry of the vault.

Recommendations

There are multiple potential solutions to this issue:

- The system could maintain a separate, different accounting structure to ensure `xDai` is allocated to the correct depositors, but this would require extensive refactoring of Gnosis vaults.
- `xDai` could be automatically processed into `GNO` upon being received, however, this would then open the possibility of a new DOS vector should the trade fail.
- The vaults could freeze new deposits if the awaiting `xDai` balance is considered large enough to warrant conversion, but this may negatively impact user experience.
- The vaults could convert `xDai` prior to processing new deposits but, as before, this could open a new DOS vector in the event the trade should fail.
- Alternatively, the team could take an off-chain approach such as operating a bot that ensures timely conversion of `xDai` to `GNO`, so long as the pending reward size is kept small, the incentive to perform this attack would be minimized.

Resolution

The issue has been acknowledged by the development team with the following comment:

> *"We believe the offchain swapper is the best option. Automated swaps suggested in other recommendations could lead to a DOS attack on the vault. The operator service will periodically swap xDAI to GNO. There will*

be a UI warning for vaults that do not perform periodic swaps. Additionally, the core team will run swappers to periodically execute the swapXdaiToGno function for the vaults with the most accumulated xDAI."

STW2-02	Uncallable `upgradeToAndCall()`		
Asset	`EigenPodOwner.sol`		
Status	**Resolved:** See Resolution		
Rating	Severity: Medium	Impact: Low	Likelihood: High

Description

Function `upgradeToAndCall()` is un-callable.

This is because function `upgradeToAndCall()` requires `msg.sender` to be a vault, as per the requirement specified in function `_authorizeUpgrade()` (which overrides `UUPSUpgradeable._authorizeUpgrade()`).

This means that the call to `EigenPodOwner.upgradeToAndCall()` has to be made from within a vault. At the time of testing, there were no identified functions in any of the related vaults that utilise this call.

Recommendations

Remove `upgradeToAndCall()` function if it is not necessary, or implement relevant vault functions to utilise it correctly.

Resolution

The issue has been fixed in commit e4523b5. The function `upgradeToAndCall()` was removed.

STW2-03	Potential OsToken Supply Inflation		
Asset	`OsToken.sol`		
Status	**Closed:** See Resolution		
Rating	Severity: Medium	Impact: High	Likelihood: Low

Description

The function `setController()` allows the `Owner` to set extra controllers (other than `OsTokenVaultController` contract) that can mint and burn tokens.

If the function is used carelessly, the added controllers could inflate the token supply or burn tokens without the owner's consent.

Recommendations

Ensure the risk and impact of this issue is understood and considered.

Resolution

The issue has been acknowledged by the development team with the following comment:

> *"The owner is the DAO contract. The addition/removal of the controller must pass a DAO vote."*

STW2-04	Users Can Block Ejection From Fox Vault		
Asset	`EthFoxVault.sol`		
Status	**Resolved:** See Resolution		
Rating	Severity: Low	Impact: Low	Likelihood: Low

Description

It is possible for an existing depositor to `EthFoxVault` to block their own ejection from the vault under certain circumstances.

If a user notices they are going to be ejected from `EthFoxVault` by watching the mempool, it is possible for them to frontrun the call to `eject()` by the vault blocklist manager with a call to `enterExitQueue()`. Then, if the vault has `_totalAssets < _totalShares` (i.e. when it has realised a net loss), it is possible for the user to leave a small quantity of shares such that the shares equate to zero assets when converted. The zero asset value will then trigger a revert on line [**201**] of `VaultEnterExit._enterExitQueue()`, which reverts the parent `eject()` call.

The impact of being able to block user ejection is limited. In addition to this, there is nothing to prevent the blocklist manager adding the user to the blocklist via calling `updateBlocklist()` instead.

Recommendations

To prevent this issue, the `eject()` function could have an additional check after line [**107**]:

```
if (convertToAssets(userShares) == 0) return;
```

This would enable the user to be blocked via `eject()` even when they have a dust amount of shares that do not correspond to any real vault assets.

Alternatively, the vault manager could monitor `eject()` transactions off-chain to ensure they did not revert and call `updateBlocklist()` with the user specified for any cases in which they did revert.

Resolution

The issue has been fixed in commit e4523b5. Additional check was added as suggested.

STW2-05	Price Feed Can Depeg From Real xDai Value		
Asset	`misc/XdaiExchange.sol`		
Status	**Closed:** See Resolution		
Rating	Severity: Low	Impact: Medium	Likelihood: Low

Description

The project makes use of the Chainlink `Dai` price feed in place of a price feed for `xDai`. If the prices of `Dai` and `xDai` are significantly different, then attempts to convert `xDai` to `GNO` will fail. If this price depeg persists over time, then it would result in depositors exiting the Gnosis vaults with less reward assets than they should receive.

While `xDai` should have the same price as `Dai`, a vulnerability confined to the Gnosis network's `xDai` may cause their prices to differ. In this event, the `XdaiExchange.sol` contract would not notice this difference and would continue to attempt `xDai` to `GNO` swaps using a minimum accepted exchange rate above the true swap rate (assuming a vulnerability in `xDai` causes it to be worth less than Dai). This would then cause all attempted trades to fail, meaning `xDai` could not be swapped for `GNO`.

This issue has been rated medium impact as, so long as the price re-pegs, then the swaps would again become possible. It is also not possible for an external actor to manipulate a price difference for profit, so long as the value of `xDai` is less than `Dai`.

Recommendations

If possible, the team should look to replace the Chainlink `Dai` price feed with a dedicated `xDai` price feed, or funding the deployment of such a price feed.

Alternatively, look at sourcing the price from other on-chain sources as a backup, such as a `xDai/USDC` pool of sufficient liquidity depth. This backup could then be compared against the reported Chainlink feed price and protective actions could be taken in the event that they differ significantly.

Another solution is to track the quantity of `xDai` owed to each depositor so that, if they leave the vault prior to it being converted to, `GNO` they can later be fairly compensated.

Resolution

The issue has been acknowledged by the development team.

STW2-06	Temporary Incorrect Reward On Zero `totalShares`		
Asset	`RewardSplitter.sol`		
Status	**Closed:** See Resolution		
Rating	Severity: Low	Impact: Low	Likelihood: Medium

Description

The testing team observed an edge case as follows:

1. The `RewardSplitter` contract receives a reward when the `totalShares` is zero.
2. The owner calls `increaseShares()` to distribute shares to Alice. At this point, Alice is the only shareholder.

When the observed case occurs, Alice's reward, as returned by function `rewardsOf()`, is zero. Therefore, the rewards received by `RewardSplitter` in step (1) are not owned by anyone.

This condition can be remedied by calling `syncRewards()` to update the `_rewardPerShare` used to calculate the rewards in `rewardsOf()`. After the call, the function `rewardsOf(Alice)` will show the correct value.

While the above case is temporary, the event may potentially cause confusion to the users if they are not aware of the behaviour.

Recommendations

Consider handling the condition when `totalShares` is zero.

Resolution

The issue has been acknowledged by the development team with the following comment:

> *"It won't break the UI as we call `syncRewards()` and `rewardsOf()` in `multicall()`."*

STW2-07	Data Type Inconsistency May Cause Precision Loss		
Asset	`VaultOsToken.sol`		
Status	**Resolved:** See Resolution		
Rating	Severity: Low	Impact: Low	Likelihood: Low

Description

On line [97], the variable `osTokenShares` is defined as `uint128`. However, `OsTokenVaultController.burnShares()` takes `uint256` as an argument.

This data type inconsistency may cause a loss of precision when the function is called on line [99].

Recommendations

Consider changing the data type of `osTokenShares` to `uint256`.

Resolution

The issue has been fixed in commit e4523b5. The data type of variable `osTokenShares` was changed from `uint128` to `uint256`.

STW2-08	Potential Arithmetic Underflow		
Asset	`KeeperRewards.sol`		
Status	**Closed:** See Resolution		
Rating	Severity: Low	Impact: Medium	Likelihood: Low

Description

The following code snippet on line [208] may cause arithmetic underflow:

```
unlockedMevDelta = params.unlockedMevReward - unlockedMevRewards[msg.sender].assets;
```

This is because `params.unlockedMevReward` is of type `uint160`, while `unlockedMevRewards[msg.sender].assets` is `uint192`.

Recommendations

Consider reviewing used data types to prevent arithmetic underflows.

Resolution

The issue has been acknowledged by the development team with the following comment.

> *"The total supply of ETH and GNO is far below `uint160`, so it realistically shouldn't be possible."*

STW2-09	Inconsistency Upon Value Update
Asset	`KeeperOracles.sol`
Status	**Closed:** See Resolution
Rating	Informational

Description

When an oracle is removed through function `removeOracle()`, the value of `totalOracles` is decremented.

This could lead to an inconsistency with the requirement set in `KeeperValidators._setValidatorsMinOracles()`, where `validatorsMinOracles` should be less than or equal to `totalOracles`.

Recommendations

Consider adding a check to ensure that the condition for `validatorsMinOracles` is still met with the new value of `totalOracles`.

Resolution

The issue has been acknowledged by the development team with the following comment:

> *"The update of oracles occurs very rarely (it hasn't happened so far). The minimum oracle values are also reviewed before any update."*

STW2-10	Potential Incorrect Use Of `MerkleProof.multiProofVerifyCalldata()`
Asset	`DepositDataRegistry.sol`
Status	**Closed:** See Resolution
Rating	Informational

Description

The function `registerValidators()` assumes that all validators in `keeperParams.validators` are to be registered/verified. This is because the leaves are constructed using indexes of the validators.

However, the `MerkleProof.multiProofVerifyCalldata()` does not require all leaves to be verified; it verifies only the leaves of the desired indexes. This behaviour causes the `indexes` input of function `registerValidators()` to be somewhat redundant and its only function is to identify how many validators are to be verified.

Recommendations

Ensure that this behaviour is understood.

If all validators are expected to be validated, then the `indexes` input can be removed. Identifying the number of validators can be done by calling a view function that determines the expected length of each validator data.

Resolution

The issue has been acknowledged by the development team with the following comment:

> *"The leaves must be sorted; otherwise, the multiproof check will fail."*

STW2-11	Potential Off-By-One Error
Asset	`KeeperRewards.sol`
Status	**Closed:** See Resolution
Rating	Informational

Description

On line [146], the value of `nonce` is incremented by one.

In doing so, it takes two update rewards before `isHarvestRequired()` returns `true`.

Recommendations

Ensure that this behaviour is expected.

The issue may not cause significant problem because function `harvest()` allows execution on the last two `rewardsRoot`.

Resolution

The issue has been acknowledged by the development team.

STW2-12	Miscellaneous General Comments
Asset	All contracts
Status	**Closed:** See Resolution
Rating	Informational

Description

This section details miscellaneous findings discovered by the testing team that do not have direct security implications:

1. **`balancerPoolId` Cannot Be Altered**

   Related Asset(s): XdaiExchange.sol

   The Gnosis vaults exchange `xDai` token rewards for `GNO` via an integration of Balancer. This integration has an immutable `balancerPoolId` variable that is specified for each trade to identify the route of the trade. In the event that Balancer upgrades the pool to avoid a bug or add new features this `balancerPoolId` may change.

   It is advised that `balancerPoolId` is made not immutable and has an admin only setter function added.

2. **Absence Of Setter Function**

   Related Asset(s): VaultFee.sol

   There is no setter for `feePercent`, which means that `feePercent` cannot be changed after the contract is deployed.

   Ensure that this behaviour is intended. If not, consider adding a setter function for `feePercent` to allow the owner to adjust the fee percentage as needed.

3. **`feePercent` Could Reach 100 Percent**

   Related Asset(s): VaultFee.sol

   When the `feePercent` is set to `_maxFeePercent()` (10,000, which represents 100%), all profits are distributed to the `feeRecipient`. This is reflected on line [**155**] of `VaultState.sol`. This means that the asset value per share will not increase, because all of the profits are converted into shares that belong to the `feeRecipient`.

   Ensure this behaviour is intended. Consider setting a reasonable maximum fee percentage that allows the asset value per share to increase over time.

4. **Potential Issues On Merkle Tree**

   Related Asset(s): CumulativeMerkleDrop.sol

   There is a potential arithmetic underflow in the `claim()` function on line [**59**]. This can occur if the `claim()` function is called twice with the same `account` as an input, and the `cumulativeAmount` in the second call is lower than in the first. This would result in a negative value for `periodAmount`.

   The `claim()` function also prevents identical leaves from being claimed twice, even though they may have different indexes in the Merkle tree. Identical leaves are defined as leaves with the same `account` and `cumulativeAmount` within the same Merkle tree. This is because identical leaves will have the same `cumulativeAmount`, resulting in a `periodAmount` of zero.

   Make sure this behaviour is understood. Consider adding checks during Merkle tree generation to prevent the described situations.

5. **Permissions For Fee Recipient(s)**

   Related Asset(s): EthPrivErc20Vault.sol, EthRestakePrivErc20Vault.sol

   The contracts `EthPrivErc20Vault` and `EthRestakePrivErc20Vault` require the owner to set proper permissions for the fee recipient address before it can transfer tokens. If the fee recipient is a `RewardSplitter` contract, then this contract cannot successfully call the `claimVaultTokens()` function. This could be inconvenient for the owner, as the fee recipient should be considered trusted and an integral part of the system. If both the vault and the `RewardSplitter` contract have the same owner, a similar argument could be made for any accounts that hold `RewardSplitter` shares.

   Consider allowing the fee recipient(s) (e.g., the `RewardSplitter` contract and its shareholders) to transfer tokens without the need for the owner to set permissions.

6. **Potential Difficulty Of Tracking Shares**

   Related Asset(s): RewardSplitter.sol

   The owner of the `RewardSplitter` contract can increase and decrease the shares of any account. However, if there are many accounts, tracking the share amount for each one would not be trivial. This is because the storage used to record this information (`_shareHolders`) is a mapping that lacks a record index.

   Consider restructuring the storage and adding a getter function to provide a more transparent view of each account's shares.

7. **Setter Lacks Boundary Check**

   Related Asset(s): XdaiExchange.sol

   The `StalePriceTimeDelta` variable has no maximum value check and so could be accidentally set very large, bypassing its intended use.

   Add a reasonable maximal value check to `setStalePriceTimeDelta()` to prevent accidental setting it to the wrong value.

8. **Custom Error Not Included In Errors Library**

   Related Asset(s): GnoGenesisVault.sol, ICumulativeMerkleDrop.sol

   The `InvalidInitialHarvest()` custom error is not stored in the `Errors.sol` library with other custom errors. Similar case also occurs on custom error `InvalidProof()` and `AlreadyClaimed()` which are stored in `ICumulativeMerkleDrop` interface instead of `Errors` library.

   It is recommended to store all custom errors in the same library for ease of future development.

9. **Function Name Consistency**

   Related Asset(s): EigenPodOwner.sol

   Function `queueWithdrawal()` calls `IEigenDelegationManager.queueWithdrawals()`. For consistency, consider renaming it to `queueWithdrawals()`.

10. **Stack Too Deep Error During Compilation**

    Related Asset(s): DepositDataRegistry.sol

    The testing team identified `Stack too deep` error during compilation.

```
Error:
Compiler run failed:
Error: Compiler error (/solidity/libsolidity/codegen/LValue.cpp:51):Stack too deep. Try compiling with `--via-ir` (cli) or
      ↪  the equivalent `viaIR: true` (standard JSON) while enabling the optimizer. Otherwise, try removing local variables.
CompilerError: Stack too deep. Try compiling with `--via-ir` (cli) or the equivalent `viaIR: true` (standard JSON) while
      ↪  enabling the optimizer. Otherwise, try removing local variables.
   --> src/stakewise/validators/DepositDataRegistry.sol:155:16:
    |
155 |         leaves[indexes[i]] = keccak256(
    |                ^^^^^^^
```

Although this issue can be resolved by activating `optimizer` and `via-ir`, compiling in this code would take more time.

It is worth noting that currently there are no identified security issues from using `via-ir`.

Make sure this behaviour is understood.

11. **Potential Difficulty In Tracking `_eigenPodOwners`**

    Related Asset(s): VaultEthRestaking.sol

    The `eigenPodOwner` addresses are stored in `_eigenPodOwners` after their creation. Unlike `_eigenPods`, `_eigenPodOwners` does not have getter function that will allow a user to track the existing `eigenPodOwner` addresses. Therefore, the only way to acquire this information is by capturing data in `EigenPodCreated` event.

    Consider adding a getter function for `_eigenPodOwners`.

12. **Potential Difficulty To Get `block.timestamp` Data**

    Related Asset(s): VaultEnterExit.sol

    The `_exitRequests` record's key is constructed by hashing information that consists of the `receiver`'s address, `block.timestamp`, and `positionTicket` as specified in line [**223-225**] as follows:

    ```
    _exitRequests[
          keccak256(abi.encode(receiver, block.timestamp, positionTicket))
        ] = exitingTickets;
    ```

    While the first and the last pieces of the information is available on the event data, the `block.timestamp` is not. It is not trivial for the user to search for information without the help of a blockchain explorer or similar tool that shows the detail of a transaction. Without knowing the correct timestamp associated with `enterExitQueue()` transaction, the user will not be able to call `claimExitedAssets()`.

    Consider adding the `block.timestamp` on the emitted event.

13. **Data Type Inconsistency In VaultEnterExit**

    Related Asset(s): VaultEnterExit.sol

    The return data type for function `getExitQueueIndex()` is `int256`. while on other functions in the contract assumes that `exitQueueIndex` is `uint256`.

    The testing team understands that this discrepancy can be intentional, as a negative value might indicate the absence of an exit queue.

    Make sure this behaviour is intended. Clearer documentation of this can help alleviate confusion.

14. **Circular Reference In EthFoxVault Creation**

    Related Asset(s): EthFoxVault.sol

    The struct `EthFoxVaultInitParams` as described in `IEthFoxVault` requires `address ownMevEscrow` as one of the inputs.

    ```
    struct EthFoxVaultInitParams {
      address admin;
      address ownMevEscrow;
      uint256 capacity;
      uint16 feePercent;
      string metadataIpfsHash;
    }
    ```

    If it is assumed that `EthFoxVault` is created using `EthVaultFactory`, then there is a circular reference between the vault contract and the escrow contract. This means that the escrow contract needs to be precomputed before calling `EthVaultFactory.createVault()`, where the escrow contract address becomes one of the inputs in `bytes calldata params`.

It is worth noting that the `ownMevEscrow` may be intended as an optional input, because if this address is set, `__VaultMev_init()` will set `_ownMevEscrow` which actually overrides shared `isOwnMevEscrow` flag set in `EthVaultFactory.createVault()`.

Consider removing `ownMevEscrow` to let the `__VaultMev_init()` requests `IEthVaultFactory(msg.sender).ownMevEscrow()` and decides whether this is own mev or shared mev.

15. **Data Type Inconsistency In `OsTokenConfig`**
    Related Asset(s): OsTokenConfig.sol
    The return values of `getConfig()` are of `uint256` type, but the `config` values are of `uint16`, as specified by `struct IOsTokenConfig.Config`.
    Consider changing the return data types to `uint16` for consistency.

16. **No Getter Functions On Private Variables**
    Related Asset(s): RewardSplitter.sol
    The variable `_totalRewards` and `_rewardPerShare` are marked as `private` without any getter functions. This makes it difficult to verify the state of the contract.
    Consider adding getter functions for these variables.

17. **Potential Inaccurate Comment**
    Related Asset(s): RewardSplitter.sol
    On line [**150**], it says: `// NB! make sure vault has balanceOf function to retrieve number of shares assigned`.
    However, the function does not call any `balanceOf`.
    Consider replacing `balanceOf` with `getShares`.

18. **Calling Function Internally**
    Related Asset(s): VaultEthStaking.sol
    On line [**48**], function `deposit()` is called instead of `_deposit()`.
    It would probably more gas efficient to call `_deposit()` directly instead of `deposit()`.

Recommendations

Ensure that the comments are understood and acknowledged, and consider implementing the suggestions above.

Resolution

The development team's responses to the raised issues above are as follows:

1. The issue has been fixed in commit e4523b5. The setter function for `balancerPoolId` was added.
2. Acknowledged.
3. Acknowledged.
4. Acknowledged with the following comment: *"The Merkle tree generator checks for uniqueness of the address."*
5. Acknowledged with the following comment: *"We don't want to whitelist any contracts/accounts on behalf of the owner. It's fine if they have to whitelist the fee recipient/splitter and keep track of all the whitelisted addresses."*

6. Acknowledged with the following comment: *"The RewardSplitter has a function* `sharesOf` *to fetch the shares of the user."*
7. Acknowledged with the following comment: *"It could be desirable to disable the stale price check. In such a case, the value could be set to type(uint128).max."*
8. Acknowledged with the following comment: *"We keep only the errors that are/can be reused.* `InvalidInitialHarvest`, `InvalidProof`, *and* `AlreadyClaimed` *are not."*
9. Acknowledged with the following comment: *"Only one withdrawal is made; we don't want to confuse the caller."*
10. Acknowledged.
11. Acknowldged with the following comment: *"The eigen pod owners can be tracked in Eigenlayer contracts. Every pod stores the owner."*
12. Acknowldged with the following comment: *"We don't want to break the event signature by adding a new parameter. The timestamp can be fetched from the subgraph or block explorer."*
13. Acknowldged with the following comment: *"There is a docstring in the interface: 'The exit queue index that should be used to claim exited assets. Returns -1 in case such an index does not exist.'"*
14. Acknowldged with the following comment: *"The* `EthFoxVault` *is not created through the factory. There is only a single instance of it deployed."*
15. Acknowldged with the following comment: *"The* `OsTokenConfig` *was refactored in."*
16. The issue has been fixed in commit e4523b5. The `totalRewards()` was made public. The `_rewardPerShare` could be misleading for the user as it is cumulative, so it was kept private.
17. The issue has been fixed in commit e4523b5. The `balanceOf` was replaced with `getShares`.
18. Acknowledged with the following comment: *"Fixing it would result in skipping blocklist/whitelist checks."*

Appendix A Test Suite

A non-exhaustive list of tests were constructed to aid this security review and are given along with this document. The `Forge` framework was used to perform these tests and the output is given below.

```
Ran 1 test for test/helpers/GnoValidatorsRegistry.sigp.t.sol:GnoValidatorsRegistryTest
[PASS] test_GnoValidatorsRegistry_read() (gas: 6293)
Suite result: ok. 1 passed; 0 failed; 0 skipped; finished in 9.27ms (73.17µs CPU time)

Ran 2 tests for test/vaults/modules/VaultEnterExitGnosis.sigp.t.sol:VaultEnterExitTestGnosisSigp
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 160, u: 12557, ~: 13657)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 2305)
Suite result: ok. 2 passed; 0 failed; 0 skipped; finished in 39.66ms (31.42ms CPU time)

Ran 3 tests for test/validators/DepositDataRegistry.sigp.t.sol:DepositDataRegistryTestSigp
[PASS] test_DepositDataRegistry_migrate(bytes32,uint256,address) (runs: 160, u: 111304, ~: 111553)
[PASS] test_DepositDataRegistry_setDepositDataManager(address) (runs: 160, u: 46683, ~: 46683)
[PASS] test_DepositDataRegistry_updateVaultState() (gas: 16067)
Suite result: ok. 3 passed; 0 failed; 0 skipped; finished in 53.52ms (44.98ms CPU time)

Ran 10 tests for test/vaults/ethereum/restake/EigenPodOwner.sigp.t.sol:EigenPodOwnerTestSigp
[PASS] test_EigenPodOwner_claimDelayedWithdrawals(uint256) (runs: 160, u: 2869, ~: 2869)
[PASS] test_EigenPodOwner_completeQueuedWithdrawal(address,uint256,uint256,uint32,uint256,bool) (runs: 160, u: 2613, ~: 2613)
[PASS] test_EigenPodOwner_delegateTo() (gas: 3009)
[PASS] test_EigenPodOwner_queueWithdrawal(uint256) (runs: 160, u: 3595, ~: 3595)
[PASS] test_EigenPodOwner_receive(uint256) (runs: 160, u: 2627, ~: 2627)
[PASS] test_EigenPodOwner_undelegate() (gas: 2591)
[PASS] test_EigenPodOwner_upgradeToAndCall() (gas: 3449)
[PASS] test_EigenPodOwner_verifyAndProcessWithdrawals() (gas: 3625)
[PASS] test_EigenPodOwner_verifyBalanceUpdates() (gas: 3185)
[PASS] test_EigenPodOwner_verifyWithdrawalCredentials() (gas: 2569)
Suite result: ok. 10 passed; 0 failed; 0 skipped; finished in 63.30ms (54.92ms CPU time)

Ran 2 tests for test/tokens/OsToken.sigp.t.sol:OsTokenTestSigp
[PASS] test_OsToken_mint_burn(address,uint256,uint256) (runs: 160, u: 72007, ~: 72398)
[PASS] test_OsToken_mint_burn_other_controller(address,uint256,uint256) (runs: 160, u: 102300, ~: 102690)
Suite result: ok. 2 passed; 0 failed; 0 skipped; finished in 73.54ms (65.03ms CPU time)

Ran 3 tests for test/vaults/modules/VaultEthRestaking.sigp.t.sol:VaultEthRestakingTestSigp
[PASS] test_VaultEthRestaking_createEigenPod() (gas: 3089)
[PASS] test_VaultEthRestaking_setRestakeOperatorsManager(address) (runs: 160, u: 7593, ~: 7593)
[PASS] test_VaultEthRestaking_setRestakeWithdrawalsManager(address) (runs: 160, u: 7637, ~: 7637)
Suite result: ok. 3 passed; 0 failed; 0 skipped; finished in 37.03ms (28.41ms CPU time)

Ran 3 tests for test/tokens/PriceFeed.sigp.t.sol:PriceFeedTestSigp
[PASS] test_PriceFeed_getRate() (gas: 16965)
[PASS] test_PriceFeed_getRate_mint_warp(uint256,uint256) (runs: 160, u: 136979, ~: 137067)
[PASS] test_PriceFeed_values(uint256) (runs: 160, u: 11888, ~: 11888)
Suite result: ok. 3 passed; 0 failed; 0 skipped; finished in 92.52ms (82.96ms CPU time)

Ran 10 tests for test/misc/RewardSplitter.sigp.t.sol:RewardSplitterTestSigp
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 3233, ~: 3233)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 3251)
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 2349)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 3036, ~: 3036)
[PASS] test_RewardSplitter_initial() (gas: 2525)
[PASS] test_RewardSplitter_rewards_decreaseShares_claimVaultTokens() (gas: 2459)
[PASS] test_RewardSplitter_rewards_decreaseShares_enterExitQueue() (gas: 3141)
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 3009)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 2481)
[PASS] test_RewardSplitter_updateVaultState() (gas: 3185)
Suite result: ok. 10 passed; 0 failed; 0 skipped; finished in 42.24ms (32.61ms CPU time)

Ran 5 tests for test/vaults/modules/VaultEthStaking.sigp.t.sol:VaultEthStakingTestSigp
[PASS] test_VaultEthStaking_deposit(address,address,address,uint256) (runs: 160, u: 3265, ~: 3265)
[PASS] test_VaultEthStaking_deposit_updateStateAndDeposit(address,uint256,address,uint256) (runs: 160, u: 11460, ~: 11460)
```

```
[PASS] test_VaultEthStaking_receive(address,uint256) (runs: 160, u: 3115, ~: 3115)
[PASS] test_VaultEthStaking_receiveFromMevEscrow(uint256) (runs: 160, u: 3126, ~: 3126)
[PASS] test_VaultEthStaking_updateStateAndDeposit(address,address,address,uint256) (runs: 160, u: 10356, ~: 10356)
Suite result: ok. 5 passed; 0 failed; 0 skipped; finished in 82.96ms (74.31ms CPU time)

Ran 5 tests for test/tokens/OsTokenConfig.sigp.t.sol:OsTokenConfigTestSigp
[PASS] test_OsTokenConfig_disableLtv(address) (runs: 160, u: 43181, ~: 43181)
[PASS] test_OsTokenConfig_getConfig() (gas: 11506)
[PASS] test_OsTokenConfig_setLiquidator(address) (runs: 160, u: 25024, ~: 25024)
[PASS] test_OsTokenConfig_setRedeemer(address) (runs: 160, u: 23679, ~: 23679)
[PASS] test_OsTokenConfig_updateConfig(uint16,uint16,uint16,uint16,uint16) (runs: 159, u: 36823, ~: 36901)
Suite result: ok. 5 passed; 0 failed; 0 skipped; finished in 114.17ms (105.06ms CPU time)

Ran 8 tests for test/misc/RewardSplitterGnosis.sigp.t.sol:RewardSplitterTestGnoSigp
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 3177, ~: 3177)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 3207)
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 2349)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 3014, ~: 3014)
[PASS] test_RewardSplitter_initial() (gas: 2503)
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 2987)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 2459)
[PASS] test_RewardSplitter_updateVaultState() (gas: 3141)
Suite result: ok. 8 passed; 0 failed; 0 skipped; finished in 49.56ms (38.63ms CPU time)

Ran 3 tests for test/vaults/modules/VaultGnoStaking.sigp.t.sol:VaultGnoStakingTest
[PASS] test_VaultGnoStaking_deposit() (gas: 3485)
[PASS] test_VaultGnoStaking_deposit_Fuzz(uint256,address,address) (runs: 160, u: 3383, ~: 3383)
[PASS] test_VaultGnoStaking_swapXdaiToGno() (gas: 2595)
Suite result: ok. 3 passed; 0 failed; 0 skipped; finished in 22.57ms (14.17ms CPU time)

Ran 1 test for test/vaults/modules/VaultAdmin.sigp.t.sol:VaultAdminTestSigp
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 3271, ~: 3274)
Suite result: ok. 1 passed; 0 failed; 0 skipped; finished in 27.86ms (19.10ms CPU time)

Ran 5 tests for test/vaults/modules/VaultOsToken.sigp.t.sol:VaultOsTokenTestSigp
[PASS] test_VaultOsToken_burnOsToken(uint256,uint256) (runs: 160, u: 2413, ~: 2413)
[PASS] test_VaultOsToken_enterExitQueue() (gas: 2636)
[PASS] test_VaultOsToken_liquidateOsToken() (gas: 3503)
[PASS] test_VaultOsToken_mintOsToken(uint256) (runs: 160, u: 2350, ~: 2350)
[PASS] test_VaultOsToken_redeemOsToken() (gas: 3525)
Suite result: ok. 5 passed; 0 failed; 0 skipped; finished in 35.55ms (26.82ms CPU time)

Ran 2 tests for test/vaults/modules/VaultBlocklist.sigp.t.sol:VaultBlocklistTestSigp
[PASS] test_VaultBlocklist_setBlocklistManager(address) (runs: 160, u: 3511, ~: 3511)
[PASS] test_VaultBlocklist_updateBlocklist(address,bool) (runs: 160, u: 3052, ~: 3052)
Suite result: ok. 2 passed; 0 failed; 0 skipped; finished in 36.50ms (28.04ms CPU time)

Ran 5 tests for test/vaults/modules/VaultOsTokenGnosis.sigp.t.sol:VaultOsTokenTestGnoSigp
[PASS] test_VaultOsToken_burnOsToken(uint256,uint256) (runs: 160, u: 2419, ~: 2419)
[PASS] test_VaultOsToken_enterExitQueue() (gas: 2636)
[PASS] test_VaultOsToken_liquidateOsToken() (gas: 3503)
[PASS] test_VaultOsToken_mintOsToken(uint256) (runs: 160, u: 2350, ~: 2350)
[PASS] test_VaultOsToken_redeemOsToken() (gas: 3525)
Suite result: ok. 5 passed; 0 failed; 0 skipped; finished in 35.45ms (26.63ms CPU time)

Ran 2 tests for test/vaults/modules/VaultEnterExit.sigp.t.sol:VaultEnterExitTestSigp
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 160, u: 12939, ~: 13713)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 2312)
Suite result: ok. 2 passed; 0 failed; 0 skipped; finished in 42.89ms (33.94ms CPU time)

Ran 6 tests for test/vaults/modules/VaultState.sigp.t.sol:VaultStateTestSigp
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 2793)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 3101)
[PASS] test_VaultState_shares(uint256[]) (runs: 160, u: 12653, ~: 13663)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 2678, ~: 2678)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 3077, ~: 3077)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 3469, ~: 3469)
Suite result: ok. 6 passed; 0 failed; 0 skipped; finished in 81.52ms (73.45ms CPU time)

Ran 8 tests for test/tokens/OsTokenVaultController.sigp.t.sol:OsTokenVaultControllerTestSigp
```

```
[PASS] test_OsTokenVaultController_mintShares_burnShares(address,uint256,uint256) (runs: 160, u: 128201, ~: 128666)
[PASS] test_OsTokenVaultController_mintShares_updateState(address,uint256,uint256) (runs: 160, u: 177803, ~: 177699)
[PASS] test_OsTokenVaultController_setAvgRewardPerSecond(uint256) (runs: 160, u: 45016, ~: 45070)
[PASS] test_OsTokenVaultController_setCapacity(uint256) (runs: 160, u: 23493, ~: 23547)
[PASS] test_OsTokenVaultController_setFeePercent(uint16) (runs: 160, u: 25747, ~: 25914)
[PASS] test_OsTokenVaultController_setKeeper(address) (runs: 160, u: 22848, ~: 22848)
[PASS] test_OsTokenVaultController_setTreasury(address) (runs: 160, u: 23413, ~: 23413)
[PASS] test_OsTokenVaultController_updateState() (gas: 7693)
Suite result: ok. 8 passed; 0 failed; 0 skipped; finished in 212.46ms (203.55ms CPU time)

Ran 4 tests for test/keeper/Keeper.sigp.t.sol:KeeperTestSigp
[PASS] test_KeeperRewards_canUpdateRewards() (gas: 3493)
[PASS] test_KeeperRewards_isHarvestRequired() (gas: 2686)
[PASS] test_KeeperRewards_setRewardsMinOracles(uint256) (runs: 160, u: 3302, ~: 3302)
[PASS] test_KeeperRewards_updateRewards() (gas: 3045)
Suite result: ok. 4 passed; 0 failed; 0 skipped; finished in 21.92ms (13.47ms CPU time)

Ran 6 tests for test/vaults/modules/VaultStateGnosis.sigp.t.sol:VaultStateTestGnoSigp
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 2793)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 3107)
[PASS] test_VaultState_shares(uint256[]) (runs: 160, u: 13034, ~: 13731)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 2684, ~: 2684)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 3080, ~: 3080)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 3478, ~: 3478)
Suite result: ok. 6 passed; 0 failed; 0 skipped; finished in 81.27ms (72.62ms CPU time)

Ran 3 tests for test/vaults/modules/VaultValidators.sigp.t.sol:VaultValidatorsTestSigp
[PASS] test_VaultValidators_collateralise_eth_vault() (gas: 2886)
[PASS] test_VaultValidators_collateralise_eth_vault_multi() (gas: 3334)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 3467, ~: 3467)
Suite result: ok. 3 passed; 0 failed; 0 skipped; finished in 21.82ms (13.65ms CPU time)

Ran 1 test for test/vaults/modules/VaultValidatorsGnosis.sigp.t.sol:VaultValidatorsTestGnoSigp
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 3423, ~: 3423)
Suite result: ok. 1 passed; 0 failed; 0 skipped; finished in 22.19ms (13.81ms CPU time)

Ran 2 tests for test/vaults/modules/VaultWhitelist.sigp.t.sol:VaultWhitelistTestSigp
[PASS] test_VaultWhitelist_setWhitelister(address) (runs: 160, u: 3027, ~: 3027)
[PASS] test_VaultWhitelist_updateWhitelist(address,bool) (runs: 160, u: 2634, ~: 2634)
Suite result: ok. 2 passed; 0 failed; 0 skipped; finished in 35.07ms (26.65ms CPU time)

Ran 3 tests for test/misc/XdaiExchange.sigp.t.sol:XdaiExchangeTestSigp
[PASS] test_XdaiExchange_setMaxSlippage(uint128) (runs: 160, u: 28711, ~: 28817)
[PASS] test_XdaiExchange_setStalePriceTimeDelta(uint128) (runs: 160, u: 24993, ~: 24993)
[PASS] test_XdaiExchange_swap(uint256) (runs: 160, u: 106179, ~: 108489)
Suite result: ok. 3 passed; 0 failed; 0 skipped; finished in 90.31ms (81.65ms CPU time)

Ran 25 tests for test/vaults/gnosis/GnoGenesisVault.sigp.t.sol:GnoGenesisVaultTest
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 23485, ~: 23721)
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 159, u: 1611677, ~: 1612828)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 840397)
[PASS] test_VaultGnoStaking_deposit() (gas: 125927)
[PASS] test_VaultGnoStaking_deposit_Fuzz(uint256,address,address) (runs: 160, u: 111559, ~: 111505)
[PASS] test_VaultGnoStaking_swapXdaiToGno() (gas: 140271)
[PASS] test_VaultOsToken_burnOsToken(uint256,uint256) (runs: 160, u: 918054, ~: 926717)
[PASS] test_VaultOsToken_enterExitQueue() (gas: 860801)
[PASS] test_VaultOsToken_liquidateOsToken() (gas: 1190473)
[PASS] test_VaultOsToken_mintOsToken(uint256) (runs: 160, u: 910481, ~: 917829)
[PASS] test_VaultOsToken_redeemOsToken() (gas: 1257849)
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 468152)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 1175307)
[PASS] test_VaultState_shares(uint256[]) (runs: 159, u: 456121, ~: 456278)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 2794, ~: 2794)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 1016240, ~: 1016345)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 176885, ~: 176968)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 51884, ~: 51884)
[PASS] test_migrate() (gas: 1202778)
[PASS] test_migrate_Fuzz(address,uint256,uint256) (runs: 160, u: 769247, ~: 769334)
[FAIL. Reason: revert: Legacy rewards allocated to new depositor] test_stealRewards_Vuln() (gas: 1333030)
[PASS] test_updateState() (gas: 565070)
```

```
[PASS] test_updateState_FirstHarvestNegative() (gas: 421107)
[PASS] test_vaultId() (gas: 13556)
[PASS] test_version() (gas: 14859)
Suite result: FAILED. 24 passed; 1 failed; 0 skipped; finished in 12.62s (12.64s CPU time)

Ran 41 tests for test/vaults/ethereum/restake/EthRestakeVault.sigp.t.sol:EthRestakeVaultTestSigp
[PASS] test_EigenPodOwner_claimDelayedWithdrawals(uint256) (runs: 160, u: 183582, ~: 183582)
[PASS] test_EigenPodOwner_completeQueuedWithdrawal(address,uint256,uint256,uint32,uint256,bool) (runs: 160, u: 189837, ~: 189837)
[PASS] test_EigenPodOwner_delegateTo() (gas: 189937)
[PASS] test_EigenPodOwner_queueWithdrawal(uint256) (runs: 160, u: 191309, ~: 191309)
[PASS] test_EigenPodOwner_receive(uint256) (runs: 160, u: 241854, ~: 241898)
[PASS] test_EigenPodOwner_undelegate() (gas: 187940)
[PASS] test_EigenPodOwner_upgradeToAndCall() (gas: 218323)
[PASS] test_EigenPodOwner_verifyAndProcessWithdrawals() (gas: 185680)
[PASS] test_EigenPodOwner_verifyBalanceUpdates() (gas: 184372)
[PASS] test_EigenPodOwner_verifyWithdrawalCredentials() (gas: 187050)
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 7992, ~: 7992)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 4526708)
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 4537709)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 2804306, ~: 2804224)
[PASS] test_RewardSplitter_initial() (gas: 2747600)
[PASS] test_RewardSplitter_rewards_decreaseShares_claimVaultTokens() (gas: 6776)
[PASS] test_RewardSplitter_rewards_decreaseShares_enterExitQueue() (gas: 4519007)
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 4476934)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 4512153)
[PASS] test_RewardSplitter_updateVaultState() (gas: 4390287)
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 23616, ~: 23831)
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 159, u: 1130902, ~: 1131838)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 785054)
[PASS] test_VaultEthStaking_deposit(address,address,address,uint256) (runs: 159, u: 70843, ~: 70753)
[PASS] test_VaultEthStaking_deposit_updateStateAndDeposit(address,uint256,address,uint256) (runs: 160, u: 1412732, ~: 1412551)
[PASS] test_VaultEthStaking_receive(address,uint256) (runs: 160, u: 70289, ~: 70196)
[PASS] test_VaultEthStaking_receiveFromMevEscrow(uint256) (runs: 160, u: 84637, ~: 84520)
[PASS] test_VaultEthStaking_updateStateAndDeposit(address,address,address,uint256) (runs: 159, u: 1360422, ~: 1360340)
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 1321587)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 2851720)
[PASS] test_VaultState_shares(uint256[]) (runs: 159, u: 301925, ~: 302955)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 1778193, ~: 1778268)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 1801350, ~: 1802712)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 134209, ~: 134250)
[PASS] test_VaultValidators_collateralise_eth_vault() (gas: 639225)
[PASS] test_VaultValidators_collateralise_eth_vault_multi() (gas: 1904146)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 52127, ~: 52127)
[PASS] test_receive(uint256) (runs: 160, u: 70542, ~: 70593)
[PASS] test_receive_eigenPodOwner(uint256) (runs: 160, u: 30852, ~: 30899)
[PASS] test_vaultId() (gas: 13578)
[PASS] test_version() (gas: 15167)
Suite result: ok. 41 passed; 0 failed; 0 skipped; finished in 18.37s (18.36s CPU time)

Ran 42 tests for test/vaults/ethereum/restake/EthRestakeBlocklistVault.sigp.t.sol:EthRestakeBlocklistVaultTestSigp
[PASS] test_EigenPodOwner_claimDelayedWithdrawals(uint256) (runs: 160, u: 183846, ~: 183846)
[PASS] test_EigenPodOwner_completeQueuedWithdrawal(address,uint256,uint256,uint32,uint256,bool) (runs: 160, u: 190123, ~: 190123)
[PASS] test_EigenPodOwner_delegateTo() (gas: 190245)
[PASS] test_EigenPodOwner_queueWithdrawal(uint256) (runs: 160, u: 191595, ~: 191595)
[PASS] test_EigenPodOwner_receive(uint256) (runs: 160, u: 286758, ~: 286810)
[PASS] test_EigenPodOwner_undelegate() (gas: 188248)
[PASS] test_EigenPodOwner_upgradeToAndCall() (gas: 218587)
[PASS] test_EigenPodOwner_verifyAndProcessWithdrawals() (gas: 185944)
[PASS] test_EigenPodOwner_verifyBalanceUpdates() (gas: 184636)
[PASS] test_EigenPodOwner_verifyWithdrawalCredentials() (gas: 187336)
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 7992, ~: 7992)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 4532590)
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 4543515)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 2809838, ~: 2809778)
[PASS] test_RewardSplitter_initial() (gas: 2753154)
[PASS] test_RewardSplitter_rewards_decreaseShares_claimVaultTokens() (gas: 6776)
[PASS] test_RewardSplitter_rewards_decreaseShares_enterExitQueue() (gas: 4524971)
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 4482740)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 4517959)
[PASS] test_RewardSplitter_updateVaultState() (gas: 4396011)
```

```
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 23596, ~: 23853)
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 159, u: 1151694, ~: 1152346)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 788001)
[PASS] test_VaultEthStaking_deposit(address,address,address,uint256) (runs: 160, u: 75392, ~: 75300)
[PASS] test_VaultEthStaking_deposit_updateStateAndDeposit(address,uint256,address,uint256) (runs: 160, u: 1417899, ~: 1417708)
[PASS] test_VaultEthStaking_receive(address,uint256) (runs: 160, u: 70332, ~: 70240)
[PASS] test_VaultEthStaking_receiveFromMevEscrow(uint256) (runs: 160, u: 84695, ~: 84630)
[PASS] test_VaultEthStaking_updateStateAndDeposit(address,address,address,uint256) (runs: 160, u: 1365002, ~: 1364906)
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 1321726)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 2854655)
[PASS] test_VaultState_shares(uint256[]) (runs: 159, u: 315678, ~: 316306)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 1781015, ~: 1781092)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 1804270, ~: 1805536)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 136301, ~: 136923)
[PASS] test_VaultValidators_collateralise_eth_vault() (gas: 641942)
[PASS] test_VaultValidators_collateralise_eth_vault_multi() (gas: 1906863)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 52193, ~: 52193)
[PASS] test_deposit(uint256) (runs: 160, u: 82138, ~: 82049)
[PASS] test_receive(uint256) (runs: 160, u: 65297, ~: 65212)
[PASS] test_receive_eigenPodOwner(uint256) (runs: 160, u: 69784, ~: 69823)
[PASS] test_vaultId() (gas: 13600)
[PASS] test_version() (gas: 15189)
Suite result: ok. 42 passed; 0 failed; 0 skipped; finished in 18.51s (18.50s CPU time)

Ran 32 tests for test/vaults/gnosis/GnoBlocklistErc20Vault.sigp.t.sol:GnoBlocklistErc20VaultTest
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 4577961, ~: 4577881)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 4624649)
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 4606767)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 2909617, ~: 2909520)
[PASS] test_RewardSplitter_initial() (gas: 2852961)
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 4545761)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 4581134)
[PASS] test_RewardSplitter_updateVaultState() (gas: 4458492)
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 23681, ~: 23963)
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 159, u: 1637347, ~: 1638539)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 835549)
[PASS] test_VaultGnoStaking_deposit() (gas: 133558)
[PASS] test_VaultGnoStaking_deposit_Fuzz(uint256,address,address) (runs: 160, u: 117103, ~: 117067)
[PASS] test_VaultGnoStaking_swapXdaiToGno() (gas: 132183)
[PASS] test_VaultOsToken_burnOsToken(uint256,uint256) (runs: 160, u: 924279, ~: 929605)
[PASS] test_VaultOsToken_enterExitQueue() (gas: 871851)
[PASS] test_VaultOsToken_liquidateOsToken() (gas: 1184343)
[PASS] test_VaultOsToken_mintOsToken(uint256) (runs: 160, u: 913054, ~: 920736)
[PASS] test_VaultOsToken_redeemOsToken() (gas: 1251761)
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 455066)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 1163820)
[PASS] test_VaultState_shares(uint256[]) (runs: 159, u: 473508, ~: 474348)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 980569, ~: 980687)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 1005038, ~: 1005171)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 178522, ~: 178648)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 52654, ~: 52654)
[PASS] test_deposit() (gas: 268505)
[PASS] test_mintOsToken() (gas: 1002575)
[FAIL. Reason: revert: Legacy rewards allocated to new depositor] test_stealRewards_Vuln() (gas: 1279288)
[PASS] test_transfer() (gas: 268360)
[PASS] test_vaultId() (gas: 13710)
[PASS] test_version() (gas: 15189)
Suite result: FAILED. 31 passed; 1 failed; 0 skipped; finished in 26.24s (26.23s CPU time)

Ran 32 tests for test/vaults/gnosis/GnoPrivErc20Vault.sigp.t.sol:GnoPrivErc20VaultTest
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 4734497, ~: 4734415)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 4679610)
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 4767285)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 3070031, ~: 3069950)
[PASS] test_RewardSplitter_initial() (gas: 2907757)
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 4600634)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 4636007)
[PASS] test_RewardSplitter_updateVaultState() (gas: 4513376)
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 23670, ~: 23985)
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 159, u: 2054078, ~: 2054932)
```

```
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 898939)
[PASS] test_VaultGnoStaking_deposit() (gas: 179940)
[PASS] test_VaultGnoStaking_deposit_Fuzz(uint256,address,address) (runs: 160, u: 223202, ~: 223085)
[PASS] test_VaultGnoStaking_swapXdaiToGno() (gas: 132227)
[PASS] test_VaultOsToken_burnOsToken(uint256,uint256) (runs: 160, u: 1039545, ~: 1043781)
[PASS] test_VaultOsToken_enterExitQueue() (gas: 1036787)
[PASS] test_VaultOsToken_liquidateOsToken() (gas: 1298526)
[PASS] test_VaultOsToken_mintOsToken(uint256) (runs: 160, u: 1027725, ~: 1034909)
[PASS] test_VaultOsToken_redeemOsToken() (gas: 1365944)
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 455154)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 1227238)
[PASS] test_VaultState_shares(uint256[]) (runs: 159, u: 686441, ~: 686460)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 1043936, ~: 1044021)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 1068375, ~: 1068501)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 241401, ~: 242007)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 52720, ~: 52720)
[PASS] test_deposit() (gas: 236880)
[PASS] test_mintOsToken() (gas: 1003742)
[FAIL. Reason: revert: Legacy rewards allocated to new depositor] test_stealRewards_Vuln() (gas: 1331669)
[PASS] test_transfer() (gas: 231743)
[PASS] test_vaultId() (gas: 13732)
[PASS] test_version() (gas: 15211)
Suite result: FAILED. 31 passed; 1 failed; 0 skipped; finished in 16.95s (28.56s CPU time)

Ran 31 tests for test/vaults/gnosis/GnoBlocklistVault.sigp.t.sol:GnoBlocklistVaultTest
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 2844739, ~: 2844739)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 4611577)
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 4596232)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 2903748, ~: 2903664)
[PASS] test_RewardSplitter_initial() (gas: 2847105)
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 4535215)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 4570588)
[PASS] test_RewardSplitter_updateVaultState() (gas: 4448354)
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 23487, ~: 23765)
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 159, u: 1601520, ~: 1601761)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 829831)
[PASS] test_VaultGnoStaking_deposit() (gas: 131867)
[PASS] test_VaultGnoStaking_deposit_Fuzz(uint256,address,address) (runs: 160, u: 115214, ~: 115182)
[PASS] test_VaultGnoStaking_swapXdaiToGno() (gas: 129915)
[PASS] test_VaultOsToken_burnOsToken(uint256,uint256) (runs: 160, u: 921454, ~: 924517)
[PASS] test_VaultOsToken_enterExitQueue() (gas: 860819)
[PASS] test_VaultOsToken_liquidateOsToken() (gas: 1176223)
[PASS] test_VaultOsToken_mintOsToken(uint256) (runs: 160, u: 908800, ~: 915660)
[PASS] test_VaultOsToken_redeemOsToken() (gas: 1243597)
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 452628)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 1158366)
[PASS] test_VaultState_shares(uint256[]) (runs: 159, u: 463601, ~: 464456)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 975789, ~: 975897)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 1000286, ~: 1000377)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 174203, ~: 174222)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 52038, ~: 52038)
[PASS] test_deposit() (gas: 261459)
[PASS] test_mintOsToken() (gas: 995559)
[FAIL. Reason: revert: Legacy rewards allocated to new depositor] test_stealRewards_Vuln() (gas: 1270354)
[PASS] test_vaultId() (gas: 13600)
[PASS] test_version() (gas: 15035)
Suite result: FAILED. 30 passed; 1 failed; 0 skipped; finished in 25.95s (25.36s CPU time)

Ran 42 tests for test/vaults/ethereum/EthErc20Vault.sigp.t.sol:EthErc20VaultTestSigp
[PASS] test_KeeperRewards_canUpdateRewards() (gas: 20654)
[PASS] test_KeeperRewards_isHarvestRequired() (gas: 2854668)
[PASS] test_KeeperRewards_setRewardsMinOracles(uint256) (runs: 160, u: 22323, ~: 21822)
[PASS] test_KeeperRewards_updateRewards() (gas: 784621)
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 4510314, ~: 4510213)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 4537538)
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 4543538)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 2804615, ~: 2804530)
[PASS] test_RewardSplitter_initial() (gas: 2747950)
[PASS] test_RewardSplitter_rewards_decreaseShares_claimVaultTokens() (gas: 4488026)
[PASS] test_RewardSplitter_rewards_decreaseShares_enterExitQueue() (gas: 4530299)
```

```
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 4482609)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 4517960)
[PASS] test_RewardSplitter_updateVaultState() (gas: 4395500)
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 23806, ~: 24051)
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 159, u: 1189037, ~: 1190294)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 790520)
[PASS] test_VaultEthStaking_deposit(address,address,address,uint256) (runs: 160, u: 72900, ~: 72812)
[PASS] test_VaultEthStaking_deposit_updateStateAndDeposit(address,uint256,address,uint256) (runs: 160, u: 1419652, ~: 1419550)
[PASS] test_VaultEthStaking_receive(address,uint256) (runs: 160, u: 69926, ~: 69820)
[PASS] test_VaultEthStaking_receiveFromMevEscrow(uint256) (runs: 160, u: 85589, ~: 85499)
[PASS] test_VaultEthStaking_updateStateAndDeposit(address,address,address,uint256) (runs: 160, u: 1365055, ~: 1364966)
[PASS] test_VaultOsToken_burnOsToken(uint256,uint256) (runs: 160, u: 840591, ~: 844836)
[PASS] test_VaultOsToken_enterExitQueue() (gas: 780746)
[PASS] test_VaultOsToken_liquidateOsToken() (gas: 1094979)
[PASS] test_VaultOsToken_mintOsToken(uint256) (runs: 160, u: 824080, ~: 831441)
[PASS] test_VaultOsToken_redeemOsToken() (gas: 1161818)
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 1324781)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 2855533)
[PASS] test_VaultState_shares(uint256[]) (runs: 159, u: 315082, ~: 316274)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 1780794, ~: 1780892)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 1803571, ~: 1805336)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 141051, ~: 141398)
[PASS] test_VaultValidators_collateralise_eth_vault() (gas: 639120)
[PASS] test_VaultValidators_collateralise_eth_vault_multi() (gas: 1899675)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 52809, ~: 52809)
[PASS] test_collateralise_x() (gas: 3894025)
[PASS] test_transfer(address,uint256,uint256) (runs: 160, u: 725866, ~: 725900)
[PASS] test_transferFrom(address,address,uint256,uint256) (runs: 160, u: 734402, ~: 735052)
[PASS] test_update_rewards() (gas: 1769175)
[PASS] test_vaultId() (gas: 13754)
[PASS] test_version() (gas: 15343)
Suite result: ok. 42 passed; 0 failed; 0 skipped; finished in 33.61s (36.93s CPU time)

Ran 41 tests for test/vaults/ethereum/EthPrivErc20Vault.sigp.t.sol:EthPrivErc20VaultTestSigp
[PASS] test_KeeperRewards_canUpdateRewards() (gas: 20632)
[PASS] test_KeeperRewards_isHarvestRequired() (gas: 2869818)
[PASS] test_KeeperRewards_setRewardsMinOracles(uint256) (runs: 160, u: 22081, ~: 21800)
[PASS] test_KeeperRewards_updateRewards() (gas: 784402)
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 4593593, ~: 4593510)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 4558745)
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 4626218)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 2887165, ~: 2887088)
[PASS] test_RewardSplitter_initial() (gas: 2768905)
[PASS] test_RewardSplitter_rewards_decreaseShares_claimVaultTokens() (gas: 4571328)
[PASS] test_RewardSplitter_rewards_decreaseShares_enterExitQueue() (gas: 4551610)
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 4503652)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 4539025)
[PASS] test_RewardSplitter_updateVaultState() (gas: 4416449)
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 23812, ~: 24073)
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 159, u: 1626916, ~: 1628013)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 828055)
[PASS] test_VaultEthStaking_deposit(address,address,address,uint256) (runs: 160, u: 191389, ~: 191291)
[PASS] test_VaultEthStaking_deposit_updateStateAndDeposit(address,uint256,address,uint256) (runs: 160, u: 1532070, ~: 1532053)
[PASS] test_VaultEthStaking_receive(address,uint256) (runs: 160, u: 135342, ~: 135249)
[PASS] test_VaultEthStaking_receiveFromMevEscrow(uint256) (runs: 160, u: 85656, ~: 85565)
[PASS] test_VaultEthStaking_updateStateAndDeposit(address,address,address,uint256) (runs: 160, u: 1478863, ~: 1478763)
[PASS] test_VaultOsToken_burnOsToken(uint256,uint256) (runs: 160, u: 866155, ~: 869618)
[PASS] test_VaultOsToken_enterExitQueue() (gas: 807897)
[PASS] test_VaultOsToken_liquidateOsToken() (gas: 1119917)
[PASS] test_VaultOsToken_mintOsToken(uint256) (runs: 160, u: 850199, ~: 856245)
[PASS] test_VaultOsToken_redeemOsToken() (gas: 1186756)
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 1324745)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 2870796)
[PASS] test_VaultState_shares(uint256[]) (runs: 159, u: 595010, ~: 595971)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 1796108, ~: 1796219)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 1818805, ~: 1820663)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 143638, ~: 144125)
[PASS] test_VaultValidators_collateralise_eth_vault() (gas: 661037)
[PASS] test_VaultValidators_collateralise_eth_vault_multi() (gas: 1921586)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 52918, ~: 52918)
```

```
[PASS] test_deposit(uint256) (runs: 160, u: 84471, ~: 84378)
[PASS] test_mintOsToken(uint256,uint256) (runs: 160, u: 853640, ~: 853552)
[PASS] test_receive(uint256) (runs: 160, u: 67171, ~: 67087)
[PASS] test_vaultId() (gas: 13798)
[PASS] test_version() (gas: 15365)
Suite result: ok. 41 passed; 0 failed; 0 skipped; finished in 14.16s (35.74s CPU time)

Ran 43 tests for test/vaults/ethereum/restake/EthRestakeBlocklistErc20Vault.sigp.t.sol:EthRestakeBlocklistErc20VaultTestSigp
[PASS] test_EigenPodOwner_claimDelayedWithdrawals(uint256) (runs: 160, u: 185298, ~: 185298)
[PASS] test_EigenPodOwner_completeQueuedWithdrawal(address,uint256,uint256,uint32,uint256,bool) (runs: 160, u: 191751, ~: 191751)
[PASS] test_EigenPodOwner_delegateTo() (gas: 191939)
[PASS] test_EigenPodOwner_queueWithdrawal(uint256) (runs: 160, u: 193267, ~: 193267)
[PASS] test_EigenPodOwner_receive(uint256) (runs: 160, u: 243555, ~: 243614)
[PASS] test_EigenPodOwner_undelegate() (gas: 189898)
[PASS] test_EigenPodOwner_upgradeToAndCall() (gas: 220061)
[PASS] test_EigenPodOwner_verifyAndProcessWithdrawals() (gas: 187418)
[PASS] test_EigenPodOwner_verifyBalanceUpdates() (gas: 186088)
[PASS] test_EigenPodOwner_verifyWithdrawalCredentials() (gas: 188964)
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 8036, ~: 8036)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 4546314)
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 4554762)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 2816168, ~: 2816094)
[PASS] test_RewardSplitter_initial() (gas: 2759415)
[PASS] test_RewardSplitter_rewards_decreaseShares_claimVaultTokens() (gas: 6798)
[PASS] test_RewardSplitter_rewards_decreaseShares_enterExitQueue() (gas: 4539113)
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 4494009)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 4529184)
[PASS] test_RewardSplitter_updateVaultState() (gas: 4406840)
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 23855, ~: 24095)
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 159, u: 1188566, ~: 1189710)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 797909)
[PASS] test_VaultEthRestaking_createEigenPod() (gas: 239247)
[PASS] test_VaultEthRestaking_setRestakeOperatorsManager(address) (runs: 160, u: 52947, ~: 52947)
[PASS] test_VaultEthRestaking_setRestakeWithdrawalsManager(address) (runs: 160, u: 50449, ~: 50449)
[PASS] test_VaultEthStaking_deposit(address,address,address,uint256) (runs: 160, u: 77562, ~: 77457)
[PASS] test_VaultEthStaking_deposit_updateStateAndDeposit(address,uint256,address,uint256) (runs: 160, u: 1424522, ~: 1424411)
[PASS] test_VaultEthStaking_receive(address,uint256) (runs: 160, u: 74385, ~: 74293)
[PASS] test_VaultEthStaking_receiveFromMevEscrow(uint256) (runs: 160, u: 85386, ~: 85301)
[PASS] test_VaultEthStaking_updateStateAndDeposit(address,address,address,uint256) (runs: 160, u: 1369257, ~: 1369160)
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 1324472)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 2860735)
[PASS] test_VaultState_shares(uint256[]) (runs: 159, u: 328174, ~: 329849)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 1786286, ~: 1786369)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 1809007, ~: 1810857)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 140818, ~: 141612)
[PASS] test_VaultValidators_collateralise_eth_vault() (gas: 644869)
[PASS] test_VaultValidators_collateralise_eth_vault_multi() (gas: 1909790)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 52831, ~: 52831)
[PASS] test_collateralise() (gas: 864494)
[PASS] test_vaultId() (gas: 13732)
[PASS] test_version() (gas: 15387)
Suite result: ok. 43 passed; 0 failed; 0 skipped; finished in 10.04s (17.68s CPU time)

Ran 34 tests for test/vaults/ethereum/custom/EthFoxVault.sigp.t.sol:EthFoxVaultTestSigp
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 7838, ~: 7838)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 4524527)
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 4535815)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 2802620, ~: 2802532)
[PASS] test_RewardSplitter_initial() (gas: 2746018)
[PASS] test_RewardSplitter_rewards_decreaseShares_claimVaultTokens() (gas: 6776)
[PASS] test_RewardSplitter_rewards_decreaseShares_enterExitQueue() (gas: 4516798)
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 4474930)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 4510259)
[PASS] test_RewardSplitter_updateVaultState() (gas: 4388311)
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 23365, ~: 23655)
[PASS] test_VaultBlocklist_setBlocklistManager(address) (runs: 160, u: 30048, ~: 30048)
[PASS] test_VaultBlocklist_updateBlocklist(address,bool) (runs: 160, u: 65113, ~: 74878)
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 159, u: 1148902, ~: 1149210)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 784142)
[PASS] test_VaultEthStaking_deposit(address,address,address,uint256) (runs: 160, u: 75170, ~: 75058)
```

```
[PASS] test_VaultEthStaking_deposit_updateStateAndDeposit(address,uint256,address,uint256) (runs: 160, u: 1417304, ~: 1417277)
[PASS] test_VaultEthStaking_receive(address,uint256) (runs: 160, u: 70207, ~: 70108)
[PASS] test_VaultEthStaking_receiveFromMevEscrow(uint256) (runs: 160, u: 84348, ~: 84267)
[PASS] test_VaultEthStaking_updateStateAndDeposit(address,address,address,uint256) (runs: 160, u: 1364762, ~: 1364650)
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 1321294)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 2850484)
[PASS] test_VaultState_shares(uint256[]) (runs: 159, u: 313236, ~: 314482)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 1777156, ~: 1777293)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 1800341, ~: 1801693)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 135651, ~: 136422)
[PASS] test_VaultValidators_collateralise_eth_vault() (gas: 638358)
[PASS] test_VaultValidators_collateralise_eth_vault_multi() (gas: 1898885)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 51819, ~: 51819)
[PASS] test_deposit(uint256) (runs: 160, u: 68675, ~: 68597)
[PASS] test_deposit_ejectUser(uint256) (runs: 160, u: 136012, ~: 135917)
[PASS] test_receive(uint256) (runs: 160, u: 65296, ~: 65212)
[PASS] test_vaultId() (gas: 13556)
[PASS] test_version() (gas: 14991)
Suite result: ok. 34 passed; 0 failed; 0 skipped; finished in 12.98s (18.93s CPU time)

Ran 40 tests for test/vaults/ethereum/EthGenesisVault.sigp.t.sol:EthGenesisVaultTest
[PASS] test_KeeperRewards_canUpdateRewards() (gas: 20632)
[PASS] test_KeeperRewards_isHarvestRequired() (gas: 2859578)
[PASS] test_KeeperRewards_setRewardsMinOracles(uint256) (runs: 160, u: 22061, ~: 21800)
[PASS] test_KeeperRewards_updateRewards() (gas: 779660)
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 7948, ~: 7948)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 4561628)
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 4569663)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 2801815, ~: 2801734)
[PASS] test_RewardSplitter_initial() (gas: 2745176)
[PASS] test_RewardSplitter_rewards_decreaseShares_claimVaultTokens() (gas: 6820)
[PASS] test_RewardSplitter_rewards_decreaseShares_enterExitQueue() (gas: 4554037)
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 4508712)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 4544085)
[PASS] test_RewardSplitter_updateVaultState() (gas: 4421955)
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 23600, ~: 23853)
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 159, u: 1157871, ~: 1158406)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 786522)
[PASS] test_VaultEthStaking_deposit(address,address,address,uint256) (runs: 160, u: 70814, ~: 70709)
[PASS] test_VaultEthStaking_deposit_updateStateAndDeposit(address,uint256,address,uint256) (runs: 160, u: 1426209, ~: 1426057)
[PASS] test_VaultEthStaking_receive(address,uint256) (runs: 160, u: 68076, ~: 67988)
[PASS] test_VaultEthStaking_receiveFromMevEscrow(uint256) (runs: 160, u: 82470, ~: 82367)
[PASS] test_VaultEthStaking_updateStateAndDeposit(address,address,address,uint256) (runs: 160, u: 1373928, ~: 1373818)
[PASS] test_VaultOsToken_burnOsToken(uint256,uint256) (runs: 160, u: 839334, ~: 842404)
[PASS] test_VaultOsToken_enterExitQueue() (gas: 772241)
[PASS] test_VaultOsToken_liquidateOsToken() (gas: 1100966)
[PASS] test_VaultOsToken_mintOsToken(uint256) (runs: 160, u: 821004, ~: 829013)
[PASS] test_VaultOsToken_redeemOsToken() (gas: 1167761)
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 1335194)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 2860024)
[PASS] test_VaultState_shares(uint256[]) (runs: 159, u: 305294, ~: 306288)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 1807642, ~: 1809431)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 1808179, ~: 1809731)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 139322, ~: 139811)
[PASS] test_VaultValidators_collateralise_eth_vault() (gas: 639181)
[PASS] test_VaultValidators_collateralise_eth_vault_multi() (gas: 1899733)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 52303, ~: 52303)
[PASS] test_migrate(address,uint256,uint256) (runs: 160, u: 696587, ~: 696680)
[PASS] test_receive(uint256) (runs: 160, u: 63149, ~: 63070)
[PASS] test_vaultId() (gas: 13666)
[PASS] test_version() (gas: 15211)
Suite result: ok. 40 passed; 0 failed; 0 skipped; finished in 11.94s (24.60s CPU time)

Ran 43 tests for test/vaults/ethereum/EthPrivVault.sigp.t.sol:EthPrivVaultTestSigp
[PASS] test_KeeperRewards_canUpdateRewards() (gas: 20676)
[PASS] test_KeeperRewards_isHarvestRequired() (gas: 2864286)
[PASS] test_KeeperRewards_setRewardsMinOracles(uint256) (runs: 160, u: 22303, ~: 21844)
[PASS] test_KeeperRewards_updateRewards() (gas: 784259)
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 7992, ~: 7992)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 4544912)
```

```
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 4614680)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 2880557, ~: 2880484)
[PASS] test_RewardSplitter_initial() (gas: 2762455)
[PASS] test_RewardSplitter_rewards_decreaseShares_claimVaultTokens() (gas: 6820)
[PASS] test_RewardSplitter_rewards_decreaseShares_enterExitQueue() (gas: 4537359)
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 4492301)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 4527652)
[PASS] test_RewardSplitter_updateVaultState() (gas: 4405550)
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 23640, ~: 23897)
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 159, u: 1588648, ~: 1589319)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 822027)
[PASS] test_VaultEthStaking_deposit(address,address,address,uint256) (runs: 160, u: 189116, ~: 189034)
[PASS] test_VaultEthStaking_deposit_updateStateAndDeposit(address,uint256,address,uint256) (runs: 160, u: 1525444, ~: 1525408)
[PASS] test_VaultEthStaking_receive(address,uint256) (runs: 160, u: 133460, ~: 133377)
[PASS] test_VaultEthStaking_receiveFromMevEscrow(uint256) (runs: 160, u: 85067, ~: 84993)
[PASS] test_VaultEthStaking_updateStateAndDeposit(address,address,address,uint256) (runs: 160, u: 1474538, ~: 1474419)
[PASS] test_VaultOsToken_burnOsToken(uint256,uint256) (runs: 160, u: 858356, ~: 864024)
[PASS] test_VaultOsToken_enterExitQueue() (gas: 796172)
[PASS] test_VaultOsToken_liquidateOsToken() (gas: 1111224)
[PASS] test_VaultOsToken_mintOsToken(uint256) (runs: 160, u: 842665, ~: 850663)
[PASS] test_VaultOsToken_redeemOsToken() (gas: 1178019)
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 1322119)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 2864824)
[PASS] test_VaultState_shares(uint256[]) (runs: 159, u: 580840, ~: 581999)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 1790842, ~: 1790940)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 1814206, ~: 1815384)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 139052, ~: 139556)
[PASS] test_VaultValidators_collateralise_eth_vault() (gas: 658098)
[PASS] test_VaultValidators_collateralise_eth_vault_multi() (gas: 1918669)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 52368, ~: 52368)
[PASS] test_VaultWhitelist_setWhitelister(address) (runs: 160, u: 29626, ~: 29626)
[PASS] test_VaultWhitelist_updateWhitelist(address,bool) (runs: 160, u: 63581, ~: 63581)
[PASS] test_deposit(uint256) (runs: 160, u: 82084, ~: 82000)
[PASS] test_mintOsToken(uint256,uint256) (runs: 160, u: 848096, ~: 848014)
[PASS] test_receive(uint256) (runs: 160, u: 65341, ~: 65259)
[PASS] test_vaultId() (gas: 13688)
[PASS] test_version() (gas: 15277)
Suite result: ok. 43 passed; 0 failed; 0 skipped; finished in 12.70s (26.47s CPU time)

Ran 45 tests for test/vaults/ethereum/restake/EthRestakePrivErc20Vault.sigp.t.sol:EthRestakePrivErc20VaultTestSigp
[PASS] test_EigenPodOwner_claimDelayedWithdrawals(uint256) (runs: 160, u: 185320, ~: 185320)
[PASS] test_EigenPodOwner_completeQueuedWithdrawal(address,uint256,uint256,uint32,uint256,bool) (runs: 160, u: 191795, ~: 191795)
[PASS] test_EigenPodOwner_delegateTo() (gas: 191961)
[PASS] test_EigenPodOwner_queueWithdrawal(uint256) (runs: 160, u: 193311, ~: 193311)
[PASS] test_EigenPodOwner_receive(uint256) (runs: 160, u: 243586, ~: 243636)
[PASS] test_EigenPodOwner_undelegate() (gas: 189920)
[PASS] test_EigenPodOwner_upgradeToAndCall() (gas: 220083)
[PASS] test_EigenPodOwner_verifyAndProcessWithdrawals() (gas: 187440)
[PASS] test_EigenPodOwner_verifyBalanceUpdates() (gas: 186110)
[PASS] test_EigenPodOwner_verifyWithdrawalCredentials() (gas: 189008)
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 4597116, ~: 4597047)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 4562035)
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 4632087)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 2893436, ~: 2893342)
[PASS] test_RewardSplitter_initial() (gas: 2774950)
[PASS] test_RewardSplitter_rewards_decreaseShares_claimVaultTokens() (gas: 4574689)
[PASS] test_RewardSplitter_rewards_decreaseShares_enterExitQueue() (gas: 4554834)
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 4509642)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 4544817)
[PASS] test_RewardSplitter_updateVaultState() (gas: 4422473)
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 23911, ~: 24139)
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 159, u: 1605395, ~: 1606272)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 830449)
[PASS] test_VaultEthRestaking_createEigenPod() (gas: 239291)
[PASS] test_VaultEthRestaking_setRestakeOperatorsManager(address) (runs: 160, u: 52902, ~: 52902)
[PASS] test_VaultEthRestaking_setRestakeWithdrawalsManager(address) (runs: 160, u: 50492, ~: 50492)
[PASS] test_VaultEthStaking_deposit(address,address,address,uint256) (runs: 160, u: 191609, ~: 191499)
[PASS] test_VaultEthStaking_deposit_updateStateAndDeposit(address,uint256,address,uint256) (runs: 160, u: 1532166, ~: 1532141)
[PASS] test_VaultEthStaking_receive(address,uint256) (runs: 160, u: 137689, ~: 137584)
[PASS] test_VaultEthStaking_receiveFromMevEscrow(uint256) (runs: 160, u: 85523, ~: 85433)
```

```
[PASS] test_VaultEthStaking_updateStateAndDeposit(address,address,address,uint256) (runs: 160, u: 1478852, ~: 1478753)
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 1324593)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 2873667)
[PASS] test_VaultState_shares(uint256[]) (runs: 159, u: 595743, ~: 596390)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 1799131, ~: 1799224)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 1821926, ~: 1823712)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 141246, ~: 141710)
[PASS] test_VaultValidators_collateralise_eth_vault() (gas: 664169)
[PASS] test_VaultValidators_collateralise_eth_vault_multi() (gas: 1929090)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 52918, ~: 52918)
[PASS] test_collateralise() (gas: 885827)
[PASS] test_deposit(uint256) (runs: 160, u: 84603, ~: 84520)
[PASS] test_receive(uint256) (runs: 160, u: 69330, ~: 69257)
[PASS] test_vaultId() (gas: 13776)
[PASS] test_version() (gas: 15431)
Suite result: ok. 45 passed; 0 failed; 0 skipped; finished in 36.23s (25.73s CPU time)

Ran 31 tests for test/vaults/gnosis/GnoPrivVault.sigp.t.sol:GnoPrivVaultTest
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 2899447, ~: 2899447)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 4666461)
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 4756486)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 3063906, ~: 3063830)
[PASS] test_RewardSplitter_initial() (gas: 2901813)
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 4590011)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 4625384)
[PASS] test_RewardSplitter_updateVaultState() (gas: 4503150)
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 23476, ~: 23787)
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 159, u: 2017847, ~: 2018674)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 893133)
[PASS] test_VaultGnoStaking_deposit() (gas: 177738)
[PASS] test_VaultGnoStaking_deposit_Fuzz(uint256,address,address) (runs: 160, u: 220673, ~: 220553)
[PASS] test_VaultGnoStaking_swapXdaiToGno() (gas: 129959)
[PASS] test_VaultOsToken_burnOsToken(uint256,uint256) (runs: 160, u: 1032099, ~: 1038517)
[PASS] test_VaultOsToken_enterExitQueue() (gas: 1025491)
[PASS] test_VaultOsToken_liquidateOsToken() (gas: 1290230)
[PASS] test_VaultOsToken_mintOsToken(uint256) (runs: 160, u: 1021984, ~: 1029657)
[PASS] test_VaultOsToken_redeemOsToken() (gas: 1357604)
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 452716)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 1221696)
[PASS] test_VaultState_shares(uint256[]) (runs: 159, u: 676259, ~: 676743)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 1039038, ~: 1039139)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 1063507, ~: 1063619)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 237008, ~: 237493)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 52104, ~: 52104)
[PASS] test_deposit() (gas: 234172)
[PASS] test_mintOsToken() (gas: 996772)
[FAIL. Reason: revert: Legacy rewards allocated to new depositor] test_stealRewards_Vuln() (gas: 1322647)
[PASS] test_vaultId() (gas: 13622)
[PASS] test_version() (gas: 15057)
Suite result: FAILED. 30 passed; 1 failed; 0 skipped; finished in 25.45s (26.67s CPU time)

Ran 29 tests for test/vaults/gnosis/GnoErc20Vault.sigp.t.sol:GnoErc20VaultTest
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 4567699, ~: 4567622)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 4618918)
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 4601161)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 2904223, ~: 2904122)
[PASS] test_RewardSplitter_initial() (gas: 2847563)
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 4540122)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 4575517)
[PASS] test_RewardSplitter_updateVaultState() (gas: 4452924)
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 23663, ~: 23941)
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 159, u: 1616431, ~: 1616992)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 832730)
[PASS] test_VaultGnoStaking_deposit() (gas: 127546)
[PASS] test_VaultGnoStaking_deposit_Fuzz(uint256,address,address) (runs: 160, u: 113425, ~: 113381)
[PASS] test_VaultGnoStaking_swapXdaiToGno() (gas: 132139)
[PASS] test_VaultOsToken_burnOsToken(uint256,uint256) (runs: 160, u: 918099, ~: 924161)
[PASS] test_VaultOsToken_enterExitQueue() (gas: 864098)
[PASS] test_VaultOsToken_liquidateOsToken() (gas: 1178584)
[PASS] test_VaultOsToken_mintOsToken(uint256) (runs: 160, u: 908275, ~: 915292)
```

```
[PASS] test_VaultOsToken_redeemOsToken() (gas: 1246002)
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 454971)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 1160985)
[PASS] test_VaultState_shares(uint256[]) (runs: 159, u: 463102, ~: 463624)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 977863, ~: 977989)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 1002367, ~: 1002469)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 175972, ~: 175981)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 52522, ~: 52522)
[FAIL. Reason: revert: Legacy rewards allocated to new depositor] test_stealRewards_Vuln() (gas: 1274023)
[PASS] test_vaultId() (gas: 13688)
[PASS] test_version() (gas: 15123)
Suite result: FAILED. 28 passed; 1 failed; 0 skipped; finished in 21.34s (28.53s CPU time)

Ran 29 tests for test/vaults/gnosis/GnoVault.sigp.t.sol:GnoVaultTest
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 2839319, ~: 2839319)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 4605880)
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 4590627)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 2898344, ~: 2898266)
[PASS] test_RewardSplitter_initial() (gas: 2841707)
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 4529588)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 4564983)
[PASS] test_RewardSplitter_updateVaultState() (gas: 4442808)
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 23498, ~: 23743)
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 159, u: 1581371, ~: 1582469)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 827012)
[PASS] test_VaultGnoStaking_deposit() (gas: 125872)
[PASS] test_VaultGnoStaking_deposit_Fuzz(uint256,address,address) (runs: 159, u: 111558, ~: 111514)
[PASS] test_VaultGnoStaking_swapXdaiToGno() (gas: 129871)
[PASS] test_VaultOsToken_burnOsToken(uint256,uint256) (runs: 160, u: 913360, ~: 919073)
[PASS] test_VaultOsToken_enterExitQueue() (gas: 853066)
[PASS] test_VaultOsToken_liquidateOsToken() (gas: 1170486)
[PASS] test_VaultOsToken_mintOsToken(uint256) (runs: 160, u: 904514, ~: 910216)
[PASS] test_VaultOsToken_redeemOsToken() (gas: 1237860)
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 452533)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 1155553)
[PASS] test_VaultState_shares(uint256[]) (runs: 159, u: 453095, ~: 453250)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 973082, ~: 973195)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 997574, ~: 997675)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 171566, ~: 171577)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 51928, ~: 51928)
[FAIL. Reason: revert: Legacy rewards allocated to new depositor] test_stealRewards_Vuln() (gas: 1265111)
[PASS] test_vaultId() (gas: 13578)
[PASS] test_version() (gas: 14991)
Suite result: FAILED. 28 passed; 1 failed; 0 skipped; finished in 36.29s (24.50s CPU time)

Ran 43 tests for test/vaults/ethereum/restake/EthRestakeErc20Vault.sigp.t.sol:EthRestakeErc20VaultTestSigp
[PASS] test_EigenPodOwner_claimDelayedWithdrawals(uint256) (runs: 160, u: 185034, ~: 185034)
[PASS] test_EigenPodOwner_completeQueuedWithdrawal(address,uint256,uint256,uint32,uint256,bool) (runs: 160, u: 191465, ~: 191465)
[PASS] test_EigenPodOwner_delegateTo() (gas: 191631)
[PASS] test_EigenPodOwner_queueWithdrawal(uint256) (runs: 160, u: 192981, ~: 192981)
[PASS] test_EigenPodOwner_receive(uint256) (runs: 160, u: 243310, ~: 243350)
[PASS] test_EigenPodOwner_undelegate() (gas: 189590)
[PASS] test_EigenPodOwner_upgradeToAndCall() (gas: 219797)
[PASS] test_EigenPodOwner_verifyAndProcessWithdrawals() (gas: 187154)
[PASS] test_EigenPodOwner_verifyBalanceUpdates() (gas: 185824)
[PASS] test_EigenPodOwner_verifyWithdrawalCredentials() (gas: 188678)
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 4513323, ~: 4513233)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 4540399)
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 4548923)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 2810648, ~: 2810540)
[PASS] test_RewardSplitter_initial() (gas: 2753861)
[PASS] test_RewardSplitter_rewards_decreaseShares_claimVaultTokens() (gas: 4490881)
[PASS] test_RewardSplitter_rewards_decreaseShares_enterExitQueue() (gas: 4533116)
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 4488170)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 4523345)
[PASS] test_RewardSplitter_updateVaultState() (gas: 4401083)
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 23804, ~: 24073)
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 159, u: 1167612, ~: 1167984)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 792780)
[PASS] test_VaultEthRestaking_createEigenPod() (gas: 239225)
```

```
[PASS] test_VaultEthRestaking_setRestakeOperatorsManager(address) (runs: 160, u: 52727, ~: 52727)
[PASS] test_VaultEthRestaking_setRestakeWithdrawalsManager(address) (runs: 160, u: 50405, ~: 50405)
[PASS] test_VaultEthStaking_deposit(address,address,address,uint256) (runs: 160, u: 73006, ~: 72910)
[PASS] test_VaultEthStaking_deposit_updateStateAndDeposit(address,uint256,address,uint256) (runs: 159, u: 1419317, ~: 1419304)
[PASS] test_VaultEthStaking_receive(address,uint256) (runs: 160, u: 72179, ~: 72067)
[PASS] test_VaultEthStaking_receiveFromMevEscrow(uint256) (runs: 160, u: 85278, ~: 85191)
[PASS] test_VaultEthStaking_updateStateAndDeposit(address,address,address,uint256) (runs: 160, u: 1364660, ~: 1364561)
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 1324300)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 2857734)
[PASS] test_VaultState_shares(uint256[]) (runs: 159, u: 315791, ~: 316606)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 1783409, ~: 1783512)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 1806219, ~: 1808000)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 138778, ~: 138939)
[PASS] test_VaultValidators_collateralise_eth_vault() (gas: 642152)
[PASS] test_VaultValidators_collateralise_eth_vault_multi() (gas: 1907073)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 52765, ~: 52765)
[PASS] test_collateralise() (gas: 861755)
[PASS] test_vaultId() (gas: 13710)
[PASS] test_version() (gas: 15365)
Suite result: ok. 43 passed; 0 failed; 0 skipped; finished in 36.29s (23.94s CPU time)

Ran 42 tests for test/vaults/ethereum/EthBlocklistErc20Vault.sigp.t.sol:EthBlocklistErc20VaultTestSigp
[PASS] test_KeeperRewards_canUpdateRewards() (gas: 20654)
[PASS] test_KeeperRewards_isHarvestRequired() (gas: 2857621)
[PASS] test_KeeperRewards_setRewardsMinOracles(uint256) (runs: 160, u: 22325, ~: 21822)
[PASS] test_KeeperRewards_updateRewards() (gas: 784698)
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 4520772, ~: 4520672)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 4543453)
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 4549377)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 2810169, ~: 2810084)
[PASS] test_RewardSplitter_initial() (gas: 2753504)
[PASS] test_RewardSplitter_rewards_decreaseShares_claimVaultTokens() (gas: 4498435)
[PASS] test_RewardSplitter_rewards_decreaseShares_enterExitQueue() (gas: 4536296)
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 4488448)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 4523799)
[PASS] test_RewardSplitter_updateVaultState() (gas: 4401257)
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 23804, ~: 24073)
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 159, u: 1209843, ~: 1210965)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 795649)
[PASS] test_VaultEthStaking_deposit(address,address,address,uint256) (runs: 160, u: 77462, ~: 77359)
[PASS] test_VaultEthStaking_deposit_updateStateAndDeposit(address,uint256,address,uint256) (runs: 160, u: 1424842, ~: 1424646)
[PASS] test_VaultEthStaking_receive(address,uint256) (runs: 160, u: 72139, ~: 72046)
[PASS] test_VaultEthStaking_receiveFromMevEscrow(uint256) (runs: 160, u: 85696, ~: 85609)
[PASS] test_VaultEthStaking_updateStateAndDeposit(address,address,address,uint256) (runs: 160, u: 1369657, ~: 1369565)
[PASS] test_VaultOsToken_burnOsToken(uint256,uint256) (runs: 160, u: 844012, ~: 850430)
[PASS] test_VaultOsToken_enterExitQueue() (gas: 788721)
[PASS] test_VaultOsToken_liquidateOsToken() (gas: 1100850)
[PASS] test_VaultOsToken_mintOsToken(uint256) (runs: 160, u: 830497, ~: 837035)
[PASS] test_VaultOsToken_redeemOsToken() (gas: 1167689)
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 1324953)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 2858534)
[PASS] test_VaultState_shares(uint256[]) (runs: 159, u: 328544, ~: 329802)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 1783651, ~: 1783749)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 1805988, ~: 1808193)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 143154, ~: 144071)
[PASS] test_VaultValidators_collateralise_eth_vault() (gas: 641837)
[PASS] test_VaultValidators_collateralise_eth_vault_multi() (gas: 1902392)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 52875, ~: 52875)
[PASS] test_collateralise_x() (gas: 3896802)
[PASS] test_transfer(address,uint256,uint256) (runs: 160, u: 735353, ~: 735740)
[PASS] test_transferFrom(address,address,uint256,uint256) (runs: 160, u: 744733, ~: 744870)
[PASS] test_update_rewards() (gas: 1769444)
[PASS] test_vaultId() (gas: 13776)
[PASS] test_version() (gas: 15365)
Suite result: ok. 42 passed; 0 failed; 0 skipped; finished in 36.29s (35.40s CPU time)

Ran 43 tests for test/vaults/ethereum/EthBlocklistVault.sigp.t.sol:EthBlocklistVaultTestSigp
[PASS] test_KeeperRewards_canUpdateRewards() (gas: 20676)
[PASS] test_KeeperRewards_isHarvestRequired() (gas: 2851420)
[PASS] test_KeeperRewards_setRewardsMinOracles(uint256) (runs: 160, u: 21991, ~: 21822)
```

```
[PASS] test_KeeperRewards_updateRewards() (gas: 784160)
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 7970, ~: 7970)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 4529389)
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 4537817)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 2803831, ~: 2803764)
[PASS] test_RewardSplitter_initial() (gas: 2747206)
[PASS] test_RewardSplitter_rewards_decreaseShares_claimVaultTokens() (gas: 6820)
[PASS] test_RewardSplitter_rewards_decreaseShares_enterExitQueue() (gas: 4521836)
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 4476888)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 4512239)
[PASS] test_RewardSplitter_updateVaultState() (gas: 4390115)
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 23604, ~: 23853)
[PASS] test_VaultBlocklist_setBlocklistManager(address) (runs: 160, u: 30312, ~: 30312)
[PASS] test_VaultBlocklist_updateBlocklist(address,bool) (runs: 160, u: 65757, ~: 75384)
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 159, u: 1172852, ~: 1173307)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 787816)
[PASS] test_VaultEthStaking_deposit(address,address,address,uint256) (runs: 160, u: 75366, ~: 75256)
[PASS] test_VaultEthStaking_deposit_updateStateAndDeposit(address,uint256,address,uint256) (runs: 160, u: 1417974, ~: 1417782)
[PASS] test_VaultEthStaking_receive(address,uint256) (runs: 160, u: 70330, ~: 70218)
[PASS] test_VaultEthStaking_receiveFromMevEscrow(uint256) (runs: 160, u: 84954, ~: 84861)
[PASS] test_VaultEthStaking_updateStateAndDeposit(address,address,address,uint256) (runs: 160, u: 1365107, ~: 1365024)
[PASS] test_VaultOsToken_burnOsToken(uint256,uint256) (runs: 160, u: 839521, ~: 844887)
[PASS] test_VaultOsToken_enterExitQueue() (gas: 777028)
[PASS] test_VaultOsToken_liquidateOsToken() (gas: 1092021)
[PASS] test_VaultOsToken_mintOsToken(uint256) (runs: 160, u: 823362, ~: 831526)
[PASS] test_VaultOsToken_redeemOsToken() (gas: 1158838)
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 1321932)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 2851892)
[PASS] test_VaultState_shares(uint256[]) (runs: 159, u: 315138, ~: 316325)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 1778050, ~: 1778151)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 1801344, ~: 1802595)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 138990, ~: 139458)
[PASS] test_VaultValidators_collateralise_eth_vault() (gas: 638930)
[PASS] test_VaultValidators_collateralise_eth_vault_multi() (gas: 1899501)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 52281, ~: 52281)
[PASS] test_deposit(uint256) (runs: 160, u: 82114, ~: 82027)
[PASS] test_mintOsToken(uint256,uint256) (runs: 160, u: 828940, ~: 828855)
[PASS] test_receive(uint256) (runs: 160, u: 65335, ~: 65256)
[PASS] test_vaultId() (gas: 13666)
[PASS] test_version() (gas: 15233)
Suite result: ok. 43 passed; 0 failed; 0 skipped; finished in 36.29s (24.01s CPU time)

Ran 39 tests for test/vaults/ethereum/EthVault.sigp.t.sol:EthVaultTestSigp
[PASS] test_KeeperRewards_canUpdateRewards() (gas: 20610)
[PASS] test_KeeperRewards_isHarvestRequired() (gas: 2848511)
[PASS] test_KeeperRewards_setRewardsMinOracles(uint256) (runs: 160, u: 22501, ~: 21756)
[PASS] test_KeeperRewards_updateRewards() (gas: 784050)
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 7904, ~: 7904)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 4523453)
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 4532012)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 2798275, ~: 2798188)
[PASS] test_RewardSplitter_initial() (gas: 2741630)
[PASS] test_RewardSplitter_rewards_decreaseShares_claimVaultTokens() (gas: 6798)
[PASS] test_RewardSplitter_rewards_decreaseShares_enterExitQueue() (gas: 4515807)
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 4471039)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 4506412)
[PASS] test_RewardSplitter_updateVaultState() (gas: 4384325)
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 23531, ~: 23809)
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 159, u: 1151853, ~: 1152827)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 782687)
[PASS] test_VaultEthStaking_deposit(address,address,address,uint256) (runs: 160, u: 70759, ~: 70643)
[PASS] test_VaultEthStaking_deposit_updateStateAndDeposit(address,uint256,address,uint256) (runs: 160, u: 1412806, ~: 1412758)
[PASS] test_VaultEthStaking_receive(address,uint256) (runs: 160, u: 68029, ~: 67926)
[PASS] test_VaultEthStaking_receiveFromMevEscrow(uint256) (runs: 160, u: 84766, ~: 84685)
[PASS] test_VaultEthStaking_updateStateAndDeposit(address,address,address,uint256) (runs: 160, u: 1360578, ~: 1360458)
[PASS] test_VaultOsToken_burnOsToken(uint256,uint256) (runs: 160, u: 831707, ~: 839296)
[PASS] test_VaultOsToken_enterExitQueue() (gas: 769031)
[PASS] test_VaultOsToken_liquidateOsToken() (gas: 1086109)
[PASS] test_VaultOsToken_mintOsToken(uint256) (runs: 160, u: 819895, ~: 825935)
[PASS] test_VaultOsToken_redeemOsToken() (gas: 1152904)
```

```
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 1321771)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 2848891)
[PASS] test_VaultState_shares(uint256[]) (runs: 159, u: 302059, ~: 302961)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 1775213, ~: 1775305)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 1798471, ~: 1799749)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 136131, ~: 136741)
[PASS] test_VaultValidators_collateralise_eth_vault() (gas: 636191)
[PASS] test_VaultValidators_collateralise_eth_vault_multi() (gas: 1896814)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 52171, ~: 52171)
[PASS] test_collateralise_eth_vault_multi() (gas: 1896591)
[PASS] test_vaultId() (gas: 13622)
[PASS] test_version() (gas: 15145)
Suite result: ok. 39 passed; 0 failed; 0 skipped; finished in 22.10s (25.01s CPU time)

Ran 43 tests for test/vaults/ethereum/restake/EthRestakePrivVault.sigp.t.sol:EthRestakePrivVaultTestSigp
[PASS] test_EigenPodOwner_claimDelayedWithdrawals(uint256) (runs: 160, u: 183868, ~: 183868)
[PASS] test_EigenPodOwner_completeQueuedWithdrawal(address,uint256,uint256,uint32,uint256,bool) (runs: 160, u: 190145, ~: 190145)
[PASS] test_EigenPodOwner_delegateTo() (gas: 190267)
[PASS] test_EigenPodOwner_queueWithdrawal(uint256) (runs: 160, u: 191661, ~: 191661)
[PASS] test_EigenPodOwner_receive(uint256) (runs: 160, u: 242128, ~: 242184)
[PASS] test_EigenPodOwner_undelegate() (gas: 188270)
[PASS] test_EigenPodOwner_upgradeToAndCall() (gas: 218631)
[PASS] test_EigenPodOwner_verifyAndProcessWithdrawals() (gas: 185988)
[PASS] test_EigenPodOwner_verifyBalanceUpdates() (gas: 184680)
[PASS] test_EigenPodOwner_verifyWithdrawalCredentials() (gas: 187380)
[PASS] test_RewardSplitter_claimVaultTokens(uint256) (runs: 160, u: 8036, ~: 8036)
[PASS] test_RewardSplitter_enterExitQueue() (gas: 4548190)
[PASS] test_RewardSplitter_increaseShares_decreaseShares_reward() (gas: 4620521)
[PASS] test_RewardSplitter_increaseShares_no_reward(uint128) (runs: 160, u: 2886760, ~: 2886674)
[PASS] test_RewardSplitter_initial() (gas: 2768491)
[PASS] test_RewardSplitter_rewards_decreaseShares_claimVaultTokens() (gas: 6776)
[PASS] test_RewardSplitter_rewards_decreaseShares_enterExitQueue() (gas: 4540571)
[PASS] test_RewardSplitter_rewards_not_lost() (gas: 4498230)
[PASS] test_RewardSplitter_rewards_temporary_lost() (gas: 4533427)
[PASS] test_RewardSplitter_updateVaultState() (gas: 4411523)
[PASS] test_VaultAdmin_setMetadata(string) (runs: 160, u: 23648, ~: 23897)
[PASS] test_VaultEnterExit_deposit_enterExitQueue_claimExitedAssets_multi(uint256[]) (runs: 159, u: 1568167, ~: 1569002)
[PASS] test_VaultEnterExit_enterExitQueue_claimExitedAssets() (gas: 824438)
[PASS] test_VaultEthStaking_deposit(address,address,address,uint256) (runs: 160, u: 189303, ~: 189188)
[PASS] test_VaultEthStaking_deposit_updateStateAndDeposit(address,uint256,address,uint256) (runs: 160, u: 1525418, ~: 1525328)
[PASS] test_VaultEthStaking_receive(address,uint256) (runs: 160, u: 135739, ~: 135647)
[PASS] test_VaultEthStaking_receiveFromMevEscrow(uint256) (runs: 160, u: 84865, ~: 84784)
[PASS] test_VaultEthStaking_updateStateAndDeposit(address,address,address,uint256) (runs: 160, u: 1474481, ~: 1474356)
[PASS] test_VaultState_isStateUpdateRequired_noValidators() (gas: 1321880)
[PASS] test_VaultState_isStateUpdateRequired_validators() (gas: 2867587)
[PASS] test_VaultState_shares(uint256[]) (runs: 159, u: 580743, ~: 582015)
[PASS] test_VaultState_updateState_own_mev(int160,uint160) (runs: 160, u: 1793798, ~: 1793892)
[PASS] test_VaultState_updateState_shared_mev(int160,uint160) (runs: 160, u: 1816951, ~: 1818336)
[PASS] test_VaultState_withdrawableAssets(uint256,uint256) (runs: 160, u: 136139, ~: 137043)
[PASS] test_VaultValidators_collateralise_eth_vault() (gas: 661154)
[PASS] test_VaultValidators_collateralise_eth_vault_multi() (gas: 1926097)
[PASS] test_VaultValidators_setValidatorsManager(address) (runs: 160, u: 52302, ~: 52302)
[PASS] test_VaultWhitelist_setWhitelister(address) (runs: 160, u: 29714, ~: 29714)
[PASS] test_VaultWhitelist_updateWhitelist(address,bool) (runs: 160, u: 64472, ~: 74649)
[PASS] test_receive(uint256) (runs: 160, u: 72858, ~: 72910)
[PASS] test_receive_eigenPodOwner(uint256) (runs: 160, u: 30871, ~: 30921)
[PASS] test_vaultId() (gas: 13622)
[PASS] test_version() (gas: 15255)
Suite result: ok. 43 passed; 0 failed; 0 skipped; finished in 36.29s (18.87s CPU time)

Ran 4 tests for test/misc/CumulativeMerkleDrop.sigp.t.sol:CumulativeMerkleDropTestSigp
[PASS] test_CumulativeMerkleDrop_claim(uint256,uint256,uint256) (runs: 160, u: 497789039, ~: 427353177)
[FAIL. Reason: panic: arithmetic underflow or overflow (0x11)] test_CumulativeMerkleDrop_claim_duplicate_account_different_amount()
      ↪  (gas: 2310839)
[PASS] test_CumulativeMerkleDrop_claim_duplicates() (gas: 2309831)
[PASS] test_CumulativeMerkleDrop_setMerkleRoot(bytes32,string) (runs: 160, u: 39966, ~: 39729)
Suite result: FAILED. 3 passed; 1 failed; 0 skipped; finished in 205.97s (205.98s CPU time)

Ran 47 test suites in 205.98s (708.03s CPU time): 889 tests passed, 8 failed, 0 skipped (897 total tests)
```

```
Failing tests:
Encountered 1 failing test in test/misc/CumulativeMerkleDrop.sigp.t.sol:CumulativeMerkleDropTestSigp
[FAIL. Reason: panic: arithmetic underflow or overflow (0x11)] test_CumulativeMerkleDrop_claim_duplicate_account_different_amount()
      ↪  (gas: 2310839)

Encountered 1 failing test in test/vaults/gnosis/GnoBlocklistErc20Vault.sigp.t.sol:GnoBlocklistErc20VaultTest
[FAIL. Reason: revert: Legacy rewards allocated to new depositor] test_stealRewards_Vuln() (gas: 1279288)

Encountered 1 failing test in test/vaults/gnosis/GnoBlocklistVault.sigp.t.sol:GnoBlocklistVaultTest
[FAIL. Reason: revert: Legacy rewards allocated to new depositor] test_stealRewards_Vuln() (gas: 1270354)

Encountered 1 failing test in test/vaults/gnosis/GnoErc20Vault.sigp.t.sol:GnoErc20VaultTest
[FAIL. Reason: revert: Legacy rewards allocated to new depositor] test_stealRewards_Vuln() (gas: 1274023)

Encountered 1 failing test in test/vaults/gnosis/GnoGenesisVault.sigp.t.sol:GnoGenesisVaultTest
[FAIL. Reason: revert: Legacy rewards allocated to new depositor] test_stealRewards_Vuln() (gas: 1333030)

Encountered 1 failing test in test/vaults/gnosis/GnoPrivErc20Vault.sigp.t.sol:GnoPrivErc20VaultTest
[FAIL. Reason: revert: Legacy rewards allocated to new depositor] test_stealRewards_Vuln() (gas: 1331669)

Encountered 1 failing test in test/vaults/gnosis/GnoPrivVault.sigp.t.sol:GnoPrivVaultTest
[FAIL. Reason: revert: Legacy rewards allocated to new depositor] test_stealRewards_Vuln() (gas: 1322647)

Encountered 1 failing test in test/vaults/gnosis/GnoVault.sigp.t.sol:GnoVaultTest
[FAIL. Reason: revert: Legacy rewards allocated to new depositor] test_stealRewards_Vuln() (gas: 1265111)

Encountered a total of 8 failing tests, 889 tests succeeded
```

Appendix B Vulnerability Severity Classification

This security review classifies vulnerabilities based on their potential impact and likelihood of occurance. The total severity of a vulnerability is derived from these two metrics based on the following matrix.

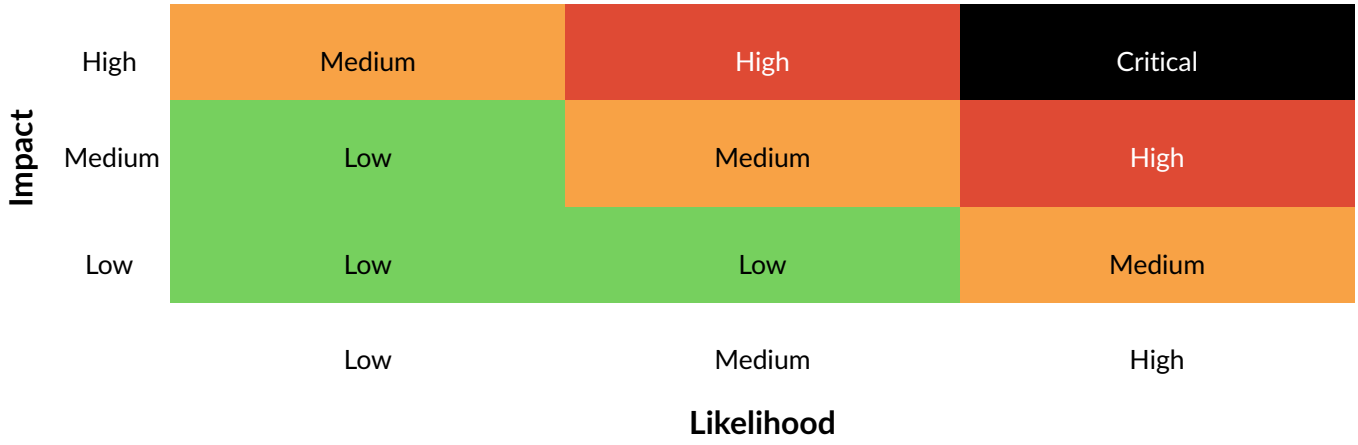

Table 1: Severity Matrix - How the severity of a vulnerability is given based on the *impact* and the *likelihood* of a vulnerability.

References

[1] Sigma Prime. Solidity Security. Blog, 2018, Available: https://blog.sigmaprime.io/solidity-security.html. [Accessed 2018].

[2] NCC Group. DASP - Top 10. Website, 2018, Available: http://www.dasp.co/. [Accessed 2018].