Vulnerability Detection Report

Report Summary

Contract Address	Source File	Detected Vulnerabilities
0x36bb138Eb36... 1d4CB244A198	contracts/PermanentPortfolioLPToken.sol	14

Structure Diagram

Detected vulnerabilities

High Severity	Medium Severity	Low Severity	Informational	Optir
0	0	14	0	(

Issues

Low(25958)

SVD-108

State Variable Default Visibility

Labeling the visibility explicitly makes it easier to catch incorrect assumptions about who can access the variable.

Variables can be specified as being `public`, `internal` or `private`. Explicitly define visibility for all state variables.

Source File

contracts/vaults/dopex/DpxArbitrumVault.sol

Locations

```
using SafeMath for uint256;
using SafeERC20 for IERC20;
error ERC4626ExceededMaxRedeem(address owner, uint
    /**
```

Low(25959)

SVD-108

State Variable Default Visibility

Labeling the visibility explicitly makes it easier to catch incorrect assumptions about who can access the variable.

Variables can be specified as being `public`, `internal` or `private`. Explicitly define visibility for all state variables.

Source File

contracts/vaults/dopex/DpxArbitrumVault.sol

Locations

```
    using SafeMath for uint256;
    using SafeERC20 for IERC20;
    error ERC4626ExceededMaxRedeem(address owner, uint

    /**
```

Low(25960)

State Variable Default Visibility

SVD-108

Labeling the visibility explicitly makes it easier to catch incorrect assumptions about who can access the variable.

Variables can be specified as being `public`, `internal` or `private`. Explicitly define visibility for all state variables.

Source File

contracts/vaults/dopex/DpxArbitrumVault.sol

Locations

```
    using SafeMath for uint256;
    using SafeERC20 for IERC20;
    error ERC4626ExceededMaxRedeem(address owner, uint

    /**
```

Low(25961)

State Variable Default Visibility

SVD-108

Labeling the visibility explicitly makes it easier to catch incorrect assumptions about who can access the variable.

Variables can be specified as being `public`, `internal` or `private`. Explicitly define visibility for all state variables.

Source File

contracts/vaults/dopex/DpxArbitrumVault.sol

Locations

```
    using SafeMath for uint256;
    using SafeERC20 for IERC20;
    error ERC4626ExceededMaxRedeem(address owner, uint

    /**
```

Low(25962)

State Variable Default Visibility

SVD-108

Labeling the visibility explicitly makes it easier to catch incorrect assumptions about who can access the variable.

Variables can be specified as being `public`, `internal` or `private`. Explicitly define visibility for all state variables.

Source File

contracts/vaults/dopex/DpxArbitrumVault.sol

Locations

```
     * @dev Attempted to deposit more assets than the 
     */
    error ERC4626ExceededMaxDeposit(
      address receiver,
      uint256 assets,
```

Low(25963)

SVD-108

State Variable Default Visibility

Labeling the visibility explicitly makes it easier to catch incorrect assumptions about who can access the variable.

Variables can be specified as being `public`, `internal` or `private`. Explicitly define visibility for all state variables.

Source File

contracts/vaults/dopex/DpxArbitrumVault.sol

Locations

```
   */
  error ERC4626ExceededMaxDeposit(
    address receiver,
    uint256 assets,
    uint256 max
```

Low(25964)

State Variable Default Visibility

SVD-108

Labeling the visibility explicitly makes it easier to catch incorrect assumptions about who can access the variable.

Variables can be specified as being `public`, `internal` or `private`. Explicitly define visibility for all state variables.

Source File

contracts/vaults/dopex/DpxArbitrumVault.sol

Locations

```
error ERC4626ExceededMaxDeposit(
  address receiver,
  uint256 assets,
  uint256 max
);
```

Low(25965)

State Variable Default Visibility

SVD-108

Labeling the visibility explicitly makes it easier to catch incorrect assumptions about who can access the variable.

Variables can be specified as being `public`, `internal` or `private`. Explicitly define visibility for all state variables.

Source File

contracts/vaults/dopex/DpxArbitrumVault.sol

Locations

```
        address receiver,
        uint256 assets,
        uint256 max
    );

```

Low(25966)

Floating Pragma

SVD-103

Contracts should be deployed with the same compiler version and flags that they have been tested with thoroughly. Locking the pragma helps to ensure that contracts do not accidentally get deployed using, for example, an outdated compiler version that might introduce bugs that affect the contract system negatively.

Lock the pragma version and also consider known bugs (https://github.com/ethereum/solidity/releases) for the compiler version that is chosen. Pragma statements can be allowed to float when a contract is intended for consumption by other developers, as in the case with contracts in a library or EthPM package. Otherwise, the developer would need to manually update the pragma in order to compile locally.

Source File

@openzeppelin/contracts/token/ERC20/ERC20.sol

Locations

```
// OpenZeppelin Contracts (last updated v4.7.0) (tok

pragma solidity ^0.8.0;
```

```
import "./IERC20.sol";
```

Low(25967) Floating Pragma

SVD-103

Contracts should be deployed with the same compiler version and flags that they have been tested with thoroughly. Locking the pragma helps to ensure that contracts do not accidentally get deployed using, for example, an outdated compiler version that might introduce bugs that affect the contract system negatively.

Lock the pragma version and also consider known bugs (https://github.com/ethereum/solidity/releases) for the compiler version that is chosen. Pragma statements can be allowed to float when a contract is intended for consumption by other developers, as in the case with contracts in a library or EthPM package. Otherwise, the developer would need to manually update the pragma in order to compile locally.

Source File

@openzeppelin/contracts/token/ERC20/utils/SafeERC20.sol

Locations

```
// OpenZeppelin Contracts (last updated v4.7.0) (tok

pragma solidity ^0.8.0;

import "../IERC20.sol";
```

‹ 1 2 › 10/page