

REPORT 62B43D9664334D0019799C72




Created	Thu Jun 23 2022 10:16:54 GMT+0000 (Coordinated Universal Time)
Number of analyses	1
User	6135edf7a6e184c5d2c6ee1e

REPORT SUMMARY

Analyses ID	Main source file	Detected vulnerabilities
19166956-1e21-4a57-a3f3-4e49615dd448	/contracts/vault.sol	3

Started	Thu Jun 23 2022 10:17:04 GMT+0000 (Coordinated Universal Time)
Finished	Thu Jun 23 2022 10:17:15 GMT+0000 (Coordinated Universal Time)
Mode	Deep
Client Tool	Mythx-Vscode-Extension
Main Source File	/Contracts/Vault.Sol

DETECTED VULNERABILITIES

 HIGH	 MEDIUM	 LOW
0	0	3

ISSUES

UNKNOWN Arithmetic operation "*" discovered
This plugin produces issues to support false positive discovery within MythX.
SWC-101

Source file
/contracts/vault.sol
Locations

```
30 | /// @dev The max amount of seconds in year.  
31 | /// accounting for leap years there are 365.25 days in year.  
32 | /// 365.25 * 86400 = 31557600.0  
33 | uint256 constant MAX_SECONDS = 31557600;
```

UNKNOWN Arithmetic operation "/" discovered
This plugin produces issues to support false positive discovery within MythX.
SWC-101

Source file
/contracts/vault.sol
Locations

```
108 | }  
109 |  
110 | /// @notice Initiates a withdrawal of vault tokens to the user.  
111 | /// @param sharesIn The amount of vault tokens to withdraw.  
112 | /// @param receiver The address to receive the vault tokens.
```

UNKNOWN Arithmetic operation "*" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/contracts/vault.sol

Locations

```
108 | }  
109 |  
110 | /// @notice Initiates a withdrawal of vault tokens to the user.  
111 | /// @param sharesIn The amount of vault tokens to withdraw.  
112 | /// @param receiver The address to receive the vault tokens.
```

UNKNOWN Arithmetic operation "/" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/contracts/vault.sol

Locations

```
129 | if (exitFee > 0) {  
130 | uint256 fee = (amountOut * exitFee) / MAX_BPS;  
131 | IERC20 wantToken.transfer(governance, fee);  
132 | amountOut = amountOut - fee;  
133 | }
```

UNKNOWN Arithmetic operation "*" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/contracts/vault.sol

Locations

```
129 | if (exitFee > 0) {  
130 | uint256 fee = (amountOut * exitFee) / MAX_BPS;  
131 | IERC20 wantToken.transfer(governance, fee);  
132 | amountOut = amountOut - fee;  
133 | }
```

UNKNOWN Arithmetic operation "/" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/contracts/vault.sol

Locations

```
135 | }  
136 |  
137 | /// @notice Calculates the total amount of underlying tokens the vault holds.  
138 | /// @return The total amount of underlying tokens the vault holds.  
139 | function totalVaultFunds() public view returns (uint256) {
```

UNKNOWN Arithmetic operation "*" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/contracts/vault.sol

Locations

```
135 | }  
136 |  
137 | /// @notice Calculates the total amount of underlying tokens the vault holds.  
138 | /// @return The total amount of underlying tokens the vault holds.  
139 | function totalVaultFunds() public view returns (uint256) {
```

UNKNOWN Arithmetic operation "-" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/contracts/vault.sol

Locations

```
136 |  
137 | /// @notice Calculates the total amount of underlying tokens the vault holds.  
138 | /// @return The total amount of underlying tokens the vault holds.  
139 | function totalVaultFunds() public view returns (uint256) {  
140 |     return  
141 |     IERC20(wantToken).balanceOf(address(this)) + totalExecutorFunds();
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file
/contracts/vault.sol
Locations

```
144 | /*/////////////////////////////////////////////////////////////////
145 | EXECUTOR DEPOSIT/WITHDRAWAL LOGIC
146 | //////////////////////////////////////////////////////////////////////////
147 |
148 | /// @notice list of trade executor's connected to vault.
149 | AddrArrayLib.Addresses tradeExecutorsList;
```

UNKNOWN Arithmetic operation "/" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file
/contracts/vault.sol
Locations

```
222 | /// @notice Emitted after exit fee updation.
223 | /// @param oldFee The old exit fee on vault.
224 | /// @param newFee The new exit fee on vault.
225 | event UpdateExitFee(uint256 oldFee, uint256 newFee);
```

UNKNOWN Arithmetic operation "/" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file
/contracts/vault.sol
Locations

```
236 | /// @notice Emitted after management fee updation.
237 | /// @param oldFee The old management fee on vault.
238 | /// @param newFee The new management fee on vault.
239 | event UpdateManagementFee(uint256 oldFee, uint256 newFee);
```

UNKNOWN Arithmetic operation "/" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/contracts/vault.sol

Locations

```
250 | /// @notice Emitted when a fees are collected.
251 | /// @param collectedFees The amount of fees collected.
252 | event FeesCollected(uint256 collectedFees);
253 |
254 | /// @notice Calculates and collects the fees from the vault.
```

UNKNOWN Arithmetic operation "-" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/contracts/vault.sol

Locations

```
267 | }
268 | if ((managementFee > 0) && (lastReportedTime < block.timestamp)) {
269 |     uint256 duration = block.timestamp - lastReportedTime;
270 |     fees +=
271 |     ((duration * managementFee * currentFunds) / MAX_SECONDS) /
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/contracts/vault.sol

Locations

```
269 | uint256 duration = block.timestamp - lastReportedTime;
270 | fees +=
271 |     (duration * managementFee * currentFunds) / MAX_SECONDS /
272 |     MAX_BPS;
273 | }
```

UNKNOWN Arithmetic operation "/" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/contracts/vault.sol

Locations

```
269 | uint256 duration = block.timestamp - lastReportedTime;
270 | fees +=
271 | ((duration * managementFee * currentFunds / MAX_SECONDS) /
272 | MAX_BPS;
273 | }
```

UNKNOWN Arithmetic operation "*" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/contracts/vault.sol

Locations

```
269 | uint256 duration = block.timestamp - lastReportedTime;
270 | fees +=
271 | ((duration * managementFee * currentFunds) / MAX_SECONDS) /
272 | MAX_BPS;
273 | }
```

UNKNOWN Arithmetic operation "-" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/contracts/vault.sol

Locations

```
273 | }
274 | if (fees > 0) {
275 |     IERC20(wantToken).safeTransfer(governance, fees);
276 |     emit FeesCollected(fees);
277 | }
278 | }
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/contracts/vault.sol

Locations

```
274 | if (fees > 0) {  
275 |     IERC20(wantToken).safeTransfer(governance, fees);  
276 |     emit FeesCollected(fees);  
277 |  
278 |  
279 |  
280 |     modifier ensureFeesAreCollected();  
281 |     collectFees();  
282 |  
283 |  
284 |     // update vault funds after fees are collected.  
285 |     prevVaultFunds = totalVaultFunds();  
286 |  
287 |     // update lastReportedTime after fees are collected.
```

UNKNOWN Arithmetic operation "/" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/contracts/vault.sol

Locations

```
278 | }  
279 |  
280 | modifier ensureFeesAreCollected();  
281 | collectFees();  
282 |  
283 | // update vault funds after fees are collected.  
284 | prevVaultFunds = totalVaultFunds();  
285 | // update lastReportedTime after fees are collected.
```

UNKNOWN Arithmetic operation "/" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/contracts/vault.sol

Locations

```
278 | }  
279 |  
280 | modifier ensureFeesAreCollected();  
281 | collectFees();  
282 |  
283 | // update vault funds after fees are collected.
```


UNKNOWN Arithmetic operation "*" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/contracts/vault.sol

Locations

```
278 | }
279 |
280 | modifier ensureFeesAreCollected() {
281 |     collectFees();
282 |     _;
283 |     // update vault funds after fees are collected.
```

UNKNOWN Arithmetic operation "*" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/contracts/vault.sol

Locations

```
278 | }
279 |
280 | modifier ensureFeesAreCollected() {
281 |     collectFees();
282 |     _;
```

UNKNOWN Arithmetic operation "++" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/contracts/vault.sol

Locations

```
355 | ).totalFunds();
356 | areFundsUpdated(blockUpdated);
357 | totalFunds += executorFunds;
358 | }
359 | return totalFunds;
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/contracts/vault.sol

Locations

```
362 | /*//////////////////////////////////////
363 | GOVERNANCE ACTIONS
364 | ////////////////////////////////////////
365 |
366 | /// @notice Emitted when a batcher is updated.
367 | /// @param oldBatcher The address of the current batcher.
368 | /// @param newBatcher The address of new batcher.
```

LOW

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.0"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

SWC-103

Source file

/contracts/vault.sol

Locations

```
1 | /// SPDX-License-Identifier: GPL-3.0-or-later
2 | pragma solidity ^0.8.0;
3 |
4 | import "@openzeppelin/contracts/token/ERC20/extensions/draft-ERC20Permit.sol";
```

LOW

State variable visibility is not set.

It is best practice to set the visibility of state variables explicitly. The default visibility for "tradeExecutorsList" is internal. Other possible visibility settings are public and private.

SWC-108

Source file

/contracts/vault.sol

Locations

```
147 |
148 | /// @notice list of trade executors connected to vault.
149 | AddrArrayLib.Addresses tradeExecutorsList;
150 |
151 | /// @notice Emitted after the vault deposits into a executor contract.
```