MythX

REPORT 628D67DDCEFD55001961E4FF

Created	Tue May 24 2022 23:18:53 GMT+0000 (Coordinated Universal Time)
Number of analyses	1
User	6197960e3494e9c8c076e89b

REPORT SUMMARY

Analyses ID	Main source file	Detected vulnerabilities
79c1a449-37f4-4bdc-80a5-43fec1a862da	Minter.sol	0

Analysis 79c1a449-37f4-4bdc-80a5-43fec1a862da MythX

Started	Tue May 24 2022 23:18:59 GMT+0000 (Coordinated Universal Time)
Finished	Tue May 24 2022 23:19:04 GMT+0000 (Coordinated Universal Time)
Mode	Deep
Client Tool	Remythx
Main Source File	`Minter.Sol`

DETECTED VULNERABILITIES

HIGH	MEDIUM	LOW
0	0	0

ISSUES

UNKNOWN Arithmetic operation "*" discovered

SWC-101

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

Source file

`Minter.sol`

Locations

```

contract Minter {
uint internal constant WEEK = 86400 * 7; // allows minting once per week (reset every Thursday 00:00 UTC)
uint internal constant EMISSION = 990;
uint internal constant TAIL_EMISSION = 2;
```

UNKNOWN Arithmetic operation "*" discovered

SWC-101

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

Source file

`Minter.sol`

Locations

```
uint public weekly = 15000000e18;
uint public active_period;
uint internal constant LOCK = 86400 * 7 * 52 * 4;

address internal initializer;
```

UNKNOWN

SWC-101

Arithmetic operation "*" discovered

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

Source file

`Minter.sol`

Locations

```
uint public weekly = 15000000e18;
uint public active_period;
uint internal constant LOCK = 86400 * 7 * 52 * 4;

address internal initializer;
```

UNKNOWN

SWC-101

Arithmetic operation "*" discovered

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

Source file

`Minter.sol`

Locations

```
uint public weekly = 15000000e18;
uint public active_period;
uint internal constant LOCK = 86400 * 7 * 52 * 4;

address internal initializer;
```

UNKNOWN

SWC-101

Arithmetic operation "*" discovered

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

Source file

`Minter.sol`

Locations

```
_ve = IVotingEscrow(__ve);
_rewards_distributor = IRewardsDistributor(__rewards_distributor);
active_period = ((block.timestamp + (2 * WEEK)) / WEEK) * WEEK;
}

```

UNKNOWN

SWC-101

Arithmetic operation "/" discovered

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

Source file

`Minter.sol`

Locations

```
_ve = IVotingEscrow(__ve);
_rewards_distributor = IRewardsDistributor(__rewards_distributor);
active_period = ((block.timestamp + (2 * WEEK)) / WEEK) * WEEK;
}

```

UNKNOWN

SWC-101

Arithmetic operation "+" discovered

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

Source file

`Minter.sol`

Locations

```
_ve = IVotingEscrow(__ve);
_rewards_distributor = IRewardsDistributor(__rewards_distributor);
active_period = ((block.timestamp + (2 * WEEK)) / WEEK) * WEEK;
}

```

UNKNOWN

SWC-101

Arithmetic operation "*" discovered

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

Source file

`Minter.sol`

Locations

```
_ve = IVotingEscrow(__ve);
_rewards_distributor = IRewardsDistributor(__rewards_distributor);
active_period = ((block.timestamp + (2 * WEEK)) / WEEK) * WEEK;
}

```

UNKNOWN

SWC-101

Arithmetic operation "++" discovered

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

Source file

Minter.sol

Locations

```
_velo.mint(address(this), max);
_velo.approve(address(_ve), type(uint).max);
for (uint i = 0; i < claimants.length; i++) {
_ve.create_lock_for(amounts[i], LOCK, claimants[i]);
}
```

UNKNOWN

SWC-101

Arithmetic operation "*" discovered

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

Source file

Minter.sol

Locations

```
}
initializer = address(0);
active_period = ((block.timestamp + WEEK) / WEEK) * WEEK;
}

```

UNKNOWN

SWC-101

Arithmetic operation "/" discovered

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

Source file

Minter.sol

Locations

```
}
initializer = address(0);
active_period = ((block.timestamp + WEEK) / WEEK) * WEEK;
}

```

UNKNOWN

SWC-101

Arithmetic operation "+" discovered

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

Source file

`Minter.sol`

Locations

```
}
initializer = address(0);
active_period = ((block.timestamp + WEEK) / WEEK) * WEEK;
}

```

UNKNOWN

SWC-101

Arithmetic operation "-" discovered

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

Source file

`Minter.sol`

Locations

```
// calculate circulating supply as total token supply - locked supply
function circulating_supply() public view returns (uint) {
return _velo.totalSupply() - _ve.totalSupply();
}

```

UNKNOWN

SWC-101

Arithmetic operation "/" discovered

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

Source file

`Minter.sol`

Locations

```
// emission calculation is 2% of available supply to mint adjusted by circulating / total supply
function calculate_emission() public view returns (uint) {
return weekly * EMISSION * circulating_supply() / PRECISION / _velo.totalSupply();
}

```

UNKNOWN

SWC-101

Arithmetic operation "/" discovered

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

Source file

`Minter.sol`

Locations

```
// emission calculation is 2% of available supply to mint adjusted by circulating / total supply
function calculate_emission() public view returns (uint) {
return weekly * EMISSION * circulating_supply() / PRECISION / _velo.totalSupply();
}

```

UNKNOWN

SWC-101

Arithmetic operation "*" discovered

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

Source file

`Minter.sol`

Locations

```
// emission calculation is 2% of available supply to mint adjusted by circulating / total supply
function calculate_emission() public view returns (uint) {
return weekly * EMISSION * circulating_supply() / PRECISION / _velo.totalSupply();
}

```

UNKNOWN

SWC-101

Arithmetic operation "*" discovered

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

Source file

`Minter.sol`

Locations

```
// emission calculation is 2% of available supply to mint adjusted by circulating / total supply
function calculate_emission() public view returns (uint) {
return weekly * EMISSION * circulating_supply() / PRECISION / _velo.totalSupply();
}

```

UNKNOWN

SWC-101

Arithmetic operation "/" discovered

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

Source file

Minter.sol

Locations

```
// calculates tail end (infinity) emissions as 0.2% of total supply
function circulating_emission() public view returns (uint) {
return (circulating_supply() * TAIL_EMISSION) / PRECISION;
}

```

UNKNOWN

SWC-101

Arithmetic operation "*" discovered

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

Source file

Minter.sol

Locations

```
// calculates tail end (infinity) emissions as 0.2% of total supply
function circulating_emission() public view returns (uint) {
return (circulating_supply() * TAIL_EMISSION) / PRECISION;
}

```

UNKNOWN

SWC-101

Arithmetic operation "/" discovered

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

Source file

Minter.sol

Locations

```
uint _veloTotal = _velo.totalSupply();
return
(((((_minted * _veTotal) / _veloTotal) * _veTotal) / _veloTotal) *
_veTotal) /
_veloTotal /
2;
}

```

UNKNOWN

SWC-101

Arithmetic operation "/" discovered

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

Source file

Minter.sol

Locations

```
uint _veloTotal = _velo.totalSupply();
return
(((((_minted * _veTotal) / _veloTotal) * _veTotal) / _veloTotal) *
_veTotal) /
_veloTotal /
2;
}
```

UNKNOWN

SWC-101

Arithmetic operation "*" discovered

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

Source file

Minter.sol

Locations

```
uint _veloTotal = _velo.totalSupply();
return
(((((_minted * _veTotal) / _veloTotal) * _veTotal) / _veloTotal) *
_veTotal) /
_veloTotal /
2;
```

UNKNOWN

SWC-101

Arithmetic operation "/" discovered

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

Source file

Minter.sol

Locations

```
uint _veloTotal = _velo.totalSupply();
return
(((((_minted * _veTotal) / _veloTotal) * _veTotal) / _veloTotal) *
_veTotal) /
_veloTotal /
```

UNKNOWN

SWC-101

Arithmetic operation "*" discovered

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

Source file

Minter.sol

Locations

```
uint _veloTotal = _velo.totalSupply();
return
((((_minted * _veTotal) / _veloTotal) * _veTotal) / _veloTotal) *
_veTotal) /
_veloTotal /
```

UNKNOWN

SWC-101

Arithmetic operation "/" discovered

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

Source file

Minter.sol

Locations

```
uint _veloTotal = _velo.totalSupply();
return
((((_minted * _veTotal) / _veloTotal) * _veTotal) / _veloTotal) *
_veTotal) /
_veloTotal /
```

UNKNOWN

SWC-101

Arithmetic operation "*" discovered

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

Source file

Minter.sol

Locations

```
uint _veloTotal = _velo.totalSupply();
return
(((((_minted * _veTotal) / _veloTotal) * _veTotal) / _veloTotal) *
_veTotal) /
_veloTotal /
```

UNKNOWN

SWC-101

Arithmetic operation "+" discovered

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

Source file

`Minter.sol`

Locations

```
function update_period() external returns (uint) {
uint _period = active_period;
if (block.timestamp >= _period + WEEK && initializer == address(0)) { // only trigger if new week
_period = (block.timestamp / WEEK) * WEEK;
active_period = _period;
```

UNKNOWN

SWC-101

Arithmetic operation "*" discovered

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

Source file

`Minter.sol`

Locations

```
uint _period = active_period;
if (block.timestamp >= _period + WEEK && initializer == address(0)) { // only trigger if new week
_period = (block.timestamp / WEEK) * WEEK;
active_period = _period;
weekly = weekly_emission();
```

UNKNOWN

SWC-101

Arithmetic operation "/" discovered

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

Source file

`Minter.sol`

Locations

```
uint _period = active_period;
if (block.timestamp >= _period + WEEK && initializer == address(0)) { // only trigger if new week
_period = (block.timestamp / WEEK) * WEEK;
active_period = _period;
weekly = weekly_emission();
```

UNKNOWN

SWC-101

Arithmetic operation "/" discovered

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

Source file

Minter.sol

Locations

```

uint _growth = calculate_growth(weekly);
uint _teamEmissions = (teamRate * (_growth + weekly)) /
(PRECISION - teamRate);
uint _required = _growth + weekly + _teamEmissions;
uint _balanceOf = _velo.balanceOf(address(this));
```

UNKNOWN

SWC-101

Arithmetic operation "*" discovered

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

Source file

Minter.sol

Locations

```

uint _growth = calculate_growth(weekly);
uint _teamEmissions = (teamRate * (_growth + weekly)) /
(PRECISION - teamRate);
uint _required = _growth + weekly + _teamEmissions;
```

UNKNOWN

SWC-101

Arithmetic operation "+" discovered

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

Source file

Minter.sol

Locations

```

uint _growth = calculate_growth(weekly);
uint _teamEmissions = (teamRate * (_growth + weekly)) /
(PRECISION - teamRate);
uint _required = _growth + weekly + _teamEmissions;
```

UNKNOWN

SWC-101

Arithmetic operation "-" discovered

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

Source file

Minter.sol

Locations

```
uint _growth = calculate_growth(weekly);
uint _teamEmissions = (teamRate * (_growth + weekly)) /
(PRECISION - teamRate);
uint _required = _growth + weekly + _teamEmissions;
uint _balanceOf = _velo.balanceOf(address(this));
```

UNKNOWN

SWC-101

Arithmetic operation "+" discovered

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

Source file

Minter.sol

Locations

```
uint _teamEmissions = (teamRate * (_growth + weekly)) /
(PRECISION - teamRate);
uint _required = _growth + weekly + _teamEmissions;
uint _balanceOf = _velo.balanceOf(address(this));
if (_balanceOf < _required) {
```

UNKNOWN

SWC-101

Arithmetic operation "+" discovered

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

Source file

Minter.sol

Locations

```
uint _teamEmissions = (teamRate * (_growth + weekly)) /
(PRECISION - teamRate);
uint _required = _growth + weekly + _teamEmissions;
uint _balanceOf = _velo.balanceOf(address(this));
if (_balanceOf < _required) {
```

UNKNOWN

SWC-101

Arithmetic operation "-" discovered

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

Source file

`Minter.sol`

Locations

```
uint _balanceOf = _velo.balanceOf(address(this));
if (_balanceOf < _required) {
_velo.mint(address(this), _required - _balanceOf);
}

```

UNKNOWN

SWC-101

Arithmetic operation "+" discovered

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

Source file

`libraries/Math.sol`

Locations

```
if (y > 3) {
z = y;
uint x = y / 2 + 1;
while (x < z) {
z = x;
```

UNKNOWN

SWC-101

Arithmetic operation "/" discovered

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

Source file

`libraries/Math.sol`

Locations

```
if (y > 3) {
z = y;
uint x = y / 2 + 1;
while (x < z) {
z = x;
```

UNKNOWN

SWC-101

Arithmetic operation "/" discovered

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

Source file

`libraries/Math.sol`

Locations

```
while (x < z) {
z = x;
x = (y / x + x) / 2;
}
} else if (y != 0) {
```

UNKNOWN

SWC-101

Arithmetic operation "+" discovered

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

Source file

`libraries/Math.sol`

Locations

```
while (x < z) {
z = x;
x = (y / x + x) / 2;
}
} else if (y != 0) {
```

UNKNOWN

SWC-101

Arithmetic operation "/" discovered

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

Source file

`libraries/Math.sol`

Locations

```
while (x < z) {
z = x;
x = (y / x + x) / 2;
}
} else if (y != 0) {
```

UNKNOWN

SWC-101

Arithmetic operation "+" discovered

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

Source file

libraries/Math.sol

Locations

```
for (uint256 y = 1 << 255; y > 0; y >>= 3) {
x <<= 1;
uint256 z = 3 * x * (x + 1) + 1;
if (n / y >= z) {
n -= y * z;
```

UNKNOWN

SWC-101

Arithmetic operation "*" discovered

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

Source file

libraries/Math.sol

Locations

```
for (uint256 y = 1 << 255; y > 0; y >>= 3) {
x <<= 1;
uint256 z = 3 * x * (x + 1) + 1;
if (n / y >= z) {
n -= y * z;
```

UNKNOWN

SWC-101

Arithmetic operation "*" discovered

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

Source file

libraries/Math.sol

Locations

```
for (uint256 y = 1 << 255; y > 0; y >>= 3) {
x <<= 1;
uint256 z = 3 * x * (x + 1) + 1;
if (n / y >= z) {
n -= y * z;
```

UNKNOWN

SWC-101

Arithmetic operation "+" discovered

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

Source file

`libraries/Math.sol`

Locations

```
for (uint256 y = 1 << 255; y > 0; y >>= 3) {
x <<= 1;
uint256 z = 3 * x * (x + 1) + 1;
if (n / y >= z) {
n -= y * z;
```

UNKNOWN

SWC-101

Arithmetic operation "/" discovered

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

Source file

`libraries/Math.sol`

Locations

```
x <<= 1;
uint256 z = 3 * x * (x + 1) + 1;
if (n / y >= z) {
n -= y * z;
x += 1;
```

UNKNOWN

SWC-101

Arithmetic operation "-=" discovered

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

Source file

`libraries/Math.sol`

Locations

```
uint256 z = 3 * x * (x + 1) + 1;
if (n / y >= z) {
n -= y * z;
x += 1;
}
```

UNKNOWN

SWC-101

Arithmetic operation "*" discovered

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

Source file

libraries/Math.sol

Locations

```
uint256 z = 3 * x * (x + 1) + 1;
if (n / y >= z) {
n -= y * z;
x += 1;
}
```

UNKNOWN

SWC-101

Arithmetic operation "+=" discovered

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

Source file

libraries/Math.sol

Locations

```
if (n / y >= z) {
n -= y * z;
x += 1;
}
}
```

UNKNOWN

SWC-110

Out of bounds array access

The index access expression can cause an exception in case of use of invalid array index value.

Source file

Minter.sol

Locations

```
_velo.approve(address(_ve), type(uint).max);
for (uint i = 0; i < claimants.length; i++) {
_ve.create_lock_for(amounts[i], LOCK, claimants[i]);
}
initializer = address(0);
```

UNKNOWN

SWC-110

Out of bounds array access

The index access expression can cause an exception in case of use of invalid array index value.

Source file

`Minter.sol`

Locations

```
_velo.approve(address(_ve), type(uint).max);
for (uint i = 0; i < claimants.length; i++) {
_ve.create_lock_for(amounts[i], LOCK, claimants[i]);
}
initializer = address(0);
```