MythX

REPORT 61F78D68923E4C0018B5A4BE

Created	Mon Jan 31 2022 07:19:04 GMT+0000 (Coordinated Universal Time)
Number of analyses	1
User	61f52e351fd393a0c51a34fe

REPORT SUMMARY

Analyses ID	Main source file	Detected vulnerabilities
697c177e-c881-4adb-8528-3082481437c6	minter.sol	0

Analysis 697c177e-c881-4adb-8528-3082481437c6 MythX

Started	Mon Jan 31 2022 07:19:11 GMT+0000 (Coordinated Universal Time)
Finished	Mon Jan 31 2022 07:19:15 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

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

SWC-101

Source file
minter.sol
Locations

```
35
36 contract BaseV1Minter {
37 uint internal constant week = 86400 * 7; // allows minting once per week (reset every Thursday 00:00 UTC)
38 uint internal constant emission = 98;
39 uint internal constant tail_emission = 2;
```

UNKNOWN Arithmetic operation "*" discovered

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

SWC-101

Source file
minter.sol
Locations

```
46 uint public weekly = 20000000e18;
47 uint public active_period;
48 uint internal constant lock = 86400 * 7 * 52 * 4;
49
50 address internal initializer;
```

UNKNOWN Arithmetic operation "*" discovered

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

SWC-101

Source file

minter.sol

Locations

```
46 uint public weekly = 20000000e18;
47 uint public active_period;
48 uint internal constant lock = 86400 * 7 * 52 * 4;
49
50 address internal initializer;
```

UNKNOWN Arithmetic operation "*" discovered

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

SWC-101

Source file

minter.sol

Locations

```
46 uint public weekly = 20000000e18;
47 uint public active_period;
48 uint internal constant lock = 86400 * 7 * 52 * 4;
49
50 address internal initializer;
```

UNKNOWN Arithmetic operation "*" discovered

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

SWC-101

Source file

minter.sol

Locations

```
62 _ve = ve(__ve);
63 _ve_dist = ve_dist(__ve_dist);
64 active_period = (block.timestamp + week) / week * week;
65 }
66
```

UNKNOWN

Arithmetic operation "/" discovered

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

SWC-101

Source file

minter.sol

Locations

```
62  _ve = ve(__ve);
63  _ve_dist = ve_dist(__ve_dist);
64  active_period = (block.timestamp + week) / week * week;
65  }
66
```

UNKNOWN

Arithmetic operation "+" discovered

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

SWC-101

Source file

minter.sol

Locations

```
62  _ve = ve(__ve);
63  _ve_dist = ve_dist(__ve_dist);
64  active_period = (block.timestamp + week) / week * week;
65  }
66
```

UNKNOWN

Arithmetic operation "++" discovered

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

SWC-101

Source file

minter.sol

Locations

```
73  _token.mint(address(this), max);
74  _token.approve(address(_ve), type(uint).max);
75  for (uint i = 0; i < claimants.length; i++) {
76  _ve.create_lock_for(amounts[i], lock, claimants[i]);
77  }
```

UNKNOWN Arithmetic operation "-" discovered

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

SWC-101

Source file

minter.sol

Locations

```
81 // calculate circulating supply as total token supply - locked supply
82 function circulating_supply() public view returns (uint) {
83 return _token.totalSupply() - _ve.totalSupply();
84 }
85
```

UNKNOWN Arithmetic operation "/" discovered

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

SWC-101

Source file

minter.sol

Locations

```
86 // emission calculation is 2% of available supply to mint adjusted by circulating / total supply
87 function calculate_emission() public view returns (uint) {
88 return weekly * emission / target_base * circulating_supply() / _token.totalSupply();
89 }
90
```

UNKNOWN Arithmetic operation "*" discovered

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

SWC-101

Source file

minter.sol

Locations

```
86 // emission calculation is 2% of available supply to mint adjusted by circulating / total supply
87 function calculate_emission() public view returns (uint) {
88 return weekly * emission / target_base * circulating_supply() / _token.totalSupply();
89 }
90
```

UNKNOWN Arithmetic operation "/" discovered

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

SWC-101

Source file

minter.sol

Locations

```
86  // emission calculation is 2% of available supply to mint adjusted by circulating / total supply
87  function calculate_emission() public view returns (uint) {
88  return weekly * emission / target_base * circulating_supply() / _token.totalSupply();
89  }
90
```

UNKNOWN Arithmetic operation "*" discovered

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

SWC-101

Source file

minter.sol

Locations

```
86  // emission calculation is 2% of available supply to mint adjusted by circulating / total supply
87  function calculate_emission() public view returns (uint) {
88  return weekly * emission / target_base * circulating_supply() / _token.totalSupply();
89  }
90
```

UNKNOWN Arithmetic operation "/" discovered

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

SWC-101

Source file

minter.sol

Locations

```
96  // calculates tail end (infinity) emissions as 0.2% of total supply
97  function circulating_emission() public view returns (uint) {
98  return circulating_supply() * tail_emission / tail_base;
99  }
100
```

UNKNOWN Arithmetic operation "*" discovered

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

SWC-101

Source file

minter.sol

Locations

```
96  // calculates tail end (infinity) emissions as 0.2% of total supply
97  function circulating_emission() public view returns (uint) {
98  return circulating_supply() * tail_emission / tail_base;
99  }
100
```

UNKNOWN Arithmetic operation "/" discovered

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

SWC-101

Source file

minter.sol

Locations

```
101 // calculate inflation and adjust ve balances accordingly
102 function calculate_growth(uint _minted) public view returns (uint) {
103 return _ve.totalSupply() * _minted / _token.totalSupply();
104 }
105
```

UNKNOWN Arithmetic operation "*" discovered

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

SWC-101

Source file

minter.sol

Locations

```
101 // calculate inflation and adjust ve balances accordingly
102 function calculate_growth(uint _minted) public view returns (uint) {
103 return _ve.totalSupply() * _minted / _token.totalSupply();
104 }
105
```

UNKNOWN

Arithmetic operation "+" discovered

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

SWC-101

Source file

minter.sol

Locations

```
107 function update_period() external returns (uint) {
108 uint _period = active_period;
109 if (block.timestamp >= _period + week) { // only trigger if new week
110 _period = block.timestamp / week * week;
111 active_period = _period;
```

UNKNOWN

Arithmetic operation "*" discovered

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

SWC-101

Source file

minter.sol

Locations

```
108 uint _period = active_period;
109 if (block.timestamp >= _period + week) { // only trigger if new week
110 _period = block.timestamp / week * week;
111 active_period = _period;
112 weekly = weekly_emission();
```

UNKNOWN

Arithmetic operation "/" discovered

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

SWC-101

Source file

minter.sol

Locations

```
108 uint _period = active_period;
109 if (block.timestamp >= _period + week) { // only trigger if new week
110 _period = block.timestamp / week * week;
111 active_period = _period;
112 weekly = weekly_emission();
```

UNKNOWN

Arithmetic operation "+" discovered

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

SWC-101

Source file

minter.sol

Locations

```
113
114 uint _growth = calculate_growth(weekly);
115 uint _required = _growth + weekly;
116 uint _balanceOf = _token.balanceOf(address(this));
117 if (_balanceOf < _required) {
```

UNKNOWN

Arithmetic operation "-" discovered

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

SWC-101

Source file

minter.sol

Locations

```
116 uint _balanceOf = _token.balanceOf(address(this));
117 if (_balanceOf < _required) {
118 _token.mint(address(this), _required-_balanceOf);
119 }
120
```

UNKNOWN

Out of bounds array access

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

SWC-110

Source file

minter.sol

Locations

```
74 _token.approve(address(_ve), type(uint).max);
75 for (uint i = 0; i < claimants.length; i++) {
76 _ve.create_lock_for(amounts[i], lock, claimants[i]);
77 }
78 initializer = address(0);
```

UNKNOWN

Out of bounds array access

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

SWC-110

Source file

minter.sol

Locations

```
74 _token.approve(address(_ve), type(uint).max);
75 for (uint i = 0; i < claimants.length; i++) {
76 _ve.create_lock_for(amounts[i], lock, claimants[i]);
77 }
78 initializer = address(0);
```