

## REPORT 60B5EFCCF1A91F0018924DE6

Created Tue Jun 01 2021 08:29:00 GMT+0000 (Coordinated Universal Time)

Number of analyses 1

User 5d48145d459fdc0011c58cba

# **REPORT SUMMARY**

Analyses ID Main source file Detected vulnerabilities

27dfddd5-60d2-4286-a326-7d5d77a493b4

PolsStake\_flat.sol

14

Started

Finished Tue Jun 01 2021 08:29:00 GMT+0000 (Coordinated Universal Time)

Mode Quick

Client Tool Mythx-Cli-0.6.22

Main Source File PolsStake\_flat.Sol

## **DETECTED VULNERABILITIES**

(HIGH	(MEDIUM	(LOW
0	4	10

### **ISSUES**

MEDIUM Function could be marked as external.

The function definition of "owner" is marked "public". However, it is never directly called by another function in the same contract or in any of its descendants. Consider to mark it as SWC-000 "external" instead.

300-000

PolsStake\_flat.sol

Locations

Source file

```
* @dev Returns the address of the current owner.

*/

function owner() public view virtual returns (address) (

return _owner.

/**
```

MEDIUM Function could be marked as external.

The function definition of "userClaimableRewardToken\_msgSender" is marked "public". However, it is never directly called by another function in the same contract or in any of its SWC-000 descendants. Consider to mark it as "external" instead.

Source file

PolsStake\_flat.sol

MEDIUM Function could be marked as external.

The function definition of "userStakedTokenUnlockTime\_msgSender" is marked "public". However, it is never directly called by another function in the same contract or in any of its descendants. Consider to mark it as "external" instead. SWC-000

PolsStake\_flat.sol

Locations

Source file

```
899
900
    function userStakedTokenUnlockTime_msgSender() public view returns (uint256 unlockTime) {
return userStakedTokenUnlockTime(msg sender);
902
904
```

MEDIUM Function could be marked as external.

The function definition of "approveStakingToken" is marked "public". However, it is never directly called by another function in the same contract or in any of its descendants. Consider to mark it as "external" instead.

Source file

PolsStake\_flat.sol

Locations

```
* https://solidity-by-example.org/delegatecall/
971
972
     function approveStakingToken(uint256 _amount) public returns (bool) {
973
974
     address(stakingToken).delegatecall(
975
                                            ess,uint256)", address(this), _amount)
976
     abi.er
977
     return success;
978
979
980
```

## LOW

SWC-103

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.

Source file

PolsStake\_flat.sol

```
5 // SPDX-License-Identifier: MIT
   pragma solidity ^0.8.0;
8
9
```

LOW

A floating pragma is set.

SWC-103

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.

Source file

PolsStake\_flat.sol

Locations



## LOW

## A floating pragma is set.

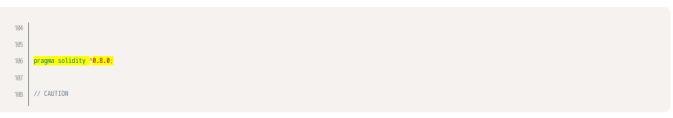
SWC-103

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.

Source file

PolsStake\_flat.sol

Locations



### LOW

#### A floating pragma is set.

SWC-103

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.

Source file

PolsStake\_flat.sol



LOW

A floating pragma is set.

SWC-103

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.

Source file

PolsStake\_flat.sol

Locations



### LOW

## A floating pragma is set.

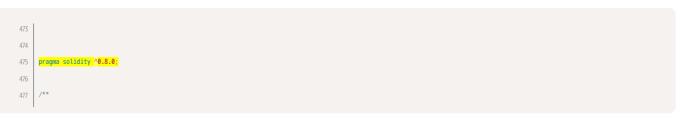
SWC-103

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.

Source file

PolsStake\_flat.sol

Locations



### LOW

#### A floating pragma is set.

SWC-103

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.

Source file

PolsStake\_flat.sol



LOW A floating pragma is set.

SWC-103

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.

Source file

PolsStake\_flat.sol

Locations



### LOW A floating pragma is set.

The current pragma Solidity directive is "">=0.7.6<0.9.0"". 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

PolsStake\_flat.sol

Locations

Source file

```
774 |
775 |
776 | pragma solidity >=0.7.6 <0.9.0 |
777 |
778 | // import "@openzeppelin/contracts/token/ERC20/IERC20.sol";
```

### LOW A floating pragma is set.

SWC-103

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.

Source file

PolsStake\_flat.sol

```
792
793
794
pragma solidity ^8.8.0
795
796 // import "@openzeppelin/contracts/utils/math/SafeCast.sol"; // OZ contracts v4
```