

REPORT 648C84E2949339001A614369

Created	Fri Jun 16 2023 15:50:58 GMT+0000 (Coordinated Universal Time)
Number of analyses	1
User	64860938a86727a2a335720f

REPORT SUMMARY

Analyses ID	Main source file	Detected vulnerabilities
c6fd3cd5-619e-42a6-be91-990c0cb4867d	contracts/DexillaExchangeV4.sol	2

Started	Fri Jun 16 2023 15:51:04 GMT+0000 (Coordinated Universal Time)
Finished	Fri Jun 16 2023 15:51:12 GMT+0000 (Coordinated Universal Time)
Mode	Deep
Client Tool	Remythx
Main Source File	Contracts/DexillaExchangeV4.Sol

DETECTED VULNERABILITIES

HIGH	MEDIUM	LOW
0	2	0

ISSUES

UNKNOWN Arithmetic operation "+" discovered

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

SWC-101

Source file

contracts/abstract/SelfPermit.sol

Locations

```
12 |  
13 | function selfPermit2(address token, uint value, uint deadline, bytes calldata signature) public payable {  
14 |     IERC20Permit2(token).permit2(msg.sender, address(this), value, deadline, signature);  
15 | }  
16 | }
```

MEDIUM Potential incorrect constructor name "selfPermit".

Until Solidity version 0.4.21 the constructor can only be defined as a function with the exact same name as the contract class. The function "selfPermit" looks similar to a constructor for "SelfPermit" but is not a constructor. Please rename to avoid confusion.

SWC-118

Source file

@openzeppelin/contracts/utils/introspection/ERC165.sol

Locations

```
8 | /**  
9 |  * @dev Implementation of the {IERC165} interface.  
10 |  *  
11 |  * Contracts that want to implement ERC165 should inherit from this contract and override {supportsInterface} to check  
12 |  * for the additional interface id that will be supported. For example:  
13 |  *  
14 |  * ``solidity  
15 |  * function supportsInterface(bytes4 interfaceId) public view virtual override returns (bool) {  
16 |  * return interfaceId == type(MyInterface).interfaceId || super.supportsInterface(interfaceId);
```