RD
AUDITORS

OVRLand Smart Contract, Code Review and Security Analysis Report

Customer: OVRLand
Prepared on: 9th February 2022
Platform: Polygon
Language: Solidity

rdauditors.com

Table of Contents

Disclaimer

This document may contain confidential information about its systems and intellectual property of the customer as well as information about potential vulnerabilities and methods of their exploitation.

The report containing confidential information can be used internally by the customer or it can be disclosed publicly after all vulnerabilities are fixed - upon the decision of the customer.

Document

Name	Smart Contract Code Review and Security Analysis Report of OVRLand
Platform	Polygon / Solidity
File 1	LightMint.sol
MD5 hash	7DF72FDB7A45BC00798194FD86A21095
SHA256 hash	7A9BE06F61E1CCF830D9B9CB8AB6339FBA8E4C8A5D650C1DDEF3840A684615D9
File 2	OVRLand.sol
MD5 hash	AB5488C77821A79F1E429D323567EB9B
SHA256 hash	F2D2882525806AD860F16A0537C2668DF7A257EF867422A89C50DD7FF529429A
File 3	OVRLandContainer.sol
MD5 hash	FB268DAE9F8AB8762F87A57061A862EE
SHA256 hash	0B36F0F9AAACB598683A8221392144A10284954230F4F165981D74151D378055
File 4	OVRMarketplace.sol

MD5 hash	731978B90DB85E7A3B52AD1A9EF5954C
SHA256 hash	9EA7DBAF4AD69C88070C7FB87BCE8FEC5DD7B5E70B6368E207512DF17A18F72B
File 5	OVRToken.sol
MD5 hash	50B3CF0F361B63BDA3E3D99F575E7E24
SHA256 hash	B9157338F664C1EC8EE88F8622D1305869D4352C49181F4C08A22550D02F8ED0
File 6	Uniswapv2router.sol
MD5 hash	77A02B2CE207D73C6D7059DDD3235506
SHA256 hash	753280666F958E5CE1A29F69AC8434EFBB8EEB949D677FB30342D2A1AC8EA634
Date	9/2/2022

Introduction

RD Auditors (Consultant) were contracted by OVRLand (Customer) to conduct a Smart Contracts Code Review and Security Analysis. This report represents the findings of the security assessment of the customer`s smart contracts and its code review conducted between 2nd February 2022 - 9th February 2022.

This contract consists of six files.

Project Scope

The scope of the project is a smart contract. We have scanned this smart contract for commonly known and more specific vulnerabilities, below are those considered (the full list includes but is not limited to):

- Reentrancy
- Timestamp Dependence
- Gas Limit and Loops
- DoS with (Unexpected) Throw
- DoS with Block Gas Limit
- Transaction-Ordering Dependence
- Byte array vulnerabilities
- Style guide violation
- Transfer forwards all gas
- ERC20 API violation
- Malicious libraries
- Compiler version not fixed
- Unchecked external call - Unchecked math
- Unsafe type inference
- Implicit visibility level

Executive Summary

According to the assessment, the customer's solidity smart contract is **well-secured.**

You are Here
🟥 Insecure
🟧 Poorly Secured
🟩 Secure
🟦 Well-Secured

Automated checks are with smartDec, Mythril, Slither and remix IDE. All issues were performed by our team, which included the analysis of code functionality, the manual audit found during automated analysis were manually reviewed and applicable vulnerabilities are presented in the audit overview section. The general overview is presented in the AS-IS section and all issues found are located in the audit overview section.

We found the following;

Total Issues	0
🟥 Critical	0
🟧 High	0
🟨 Medium	0
🟩 Low	0
🟦 Very Low	0

Code Quality

Please note that within this report SafeMath, ERC1155, Ownable, Counters, Strings, VRFConsumerBase are taken from the popular OpenZeppelin library.

The libraries within this smart contract are part of a logical algorithm. A library is a different type of smart contract that contains reusable code. Once deployed on the blockchain (only once), it is assigned to a specific address and its properties/methods can be reused many times by other contracts.

The OVRLand team have provided scenario and unit test scripts, which would help to determine the integrity of the code in an automated way.

Documentation

We were given the OVRLand source code as a Github link:

https://github.com/OVR-Platform/polygon-smart-contracts

The hash of that file is mentioned in the table. As mentioned above, It's recommended to write comments in the smart contract code, so anyone can quickly understand the programming flow as well as complex code logic.

Comments are very helpful in understanding the overall architecture of the protocol. It also provides a clear overview of the system components, including helpful details, like the lifetime of the background script.

Use of Dependencies

As per our observation, the libraries are used in this smart contract infrastructure. Those were based on well known industry standard open source projects and even core code blocks that are written well and systematically.

AS-IS Overview

LightMint

File And Function Level Report

File:	LightMint.sol
Contract:	LightMint
Import	AccessControl, Pausable, MarkleProof
Inherit	AccessControl, Pausable
Observation:	Passed
Test Report:	Passed

Sl.	Function	Type	Observation	Test Report	Conclusion	Score
1	addAdminRole	write	Passed	All Passed	No Issue	Passed
2	setOVRLand	write	Passed	All Passed	No Issue	Passed
3	setMerkleRoot	write	Passed	All Passed	No Issue	Passed
4	isClaimed	read	Passed	All Passed	No Issue	Passed
5	_setClaimed	write	Passed	All Passed	No Issue	Passed
6	claim	write	Passed	All Passed	No Issue	Passed
7	pause	write	Passed	All Passed	No Issue	Passed
8	unpause	write	Passed	All Passed	No Issue	Passed

OVRLand

File And Function Level Report

File:	OVRLand.sol
Contract:	OVRLand
Observation:	Passed
Test Report:	Passed

Sl.	Function	Type	Observation	Test Report	Conclusion	Score
1	addURIEditor	write	Passed	All Passed	No Issue	Passed
2	removeURIEditor	write	Passed	All Passed	No Issue	Passed
3	addMinter	write	Passed	All Passed	No Issue	Passed
4	removeMinter	write	Passed	All Passed	No Issue	Passed
5	addBurner	write	Passed	All Passed	No Issue	Passed
6	removeBurner	write	Passed	All Passed	No Issue	Passed
7	addAdminRole	write	Passed	All Passed	No Issue	Passed
8	removeAdminRole	write	Passed	All Passed	No Issue	Passed
9	safeMint	write	Passed	All Passed	No Issue	Passed
10	setOVRLandURI	write	Passed	All Passed	No Issue	Passed
11	burn	write	Passed	All Passed	No Issue	Passed
12	batchBurn	write	Passed	All Passed	No Issue	Passed
13	mint	write	Passed	All Passed	No Issue	Passed
14	batchMintLands	write	Passed	All Passed	No Issue	Passed

15	batchMintLandsWithUri	write	Passed	All Passed	No Issue	Passed
16	batchSetOVRLandURI	write	Passed	All Passed	No Issue	Passed
17	_burn	internal	Passed	All Passed	No Issue	Passed
18	tokenURI	read	Passed	All Passed	No Issue	Passed
19	supportsInterface	read	Passed	All Passed	No Issue	Passed

OVRLandContainer

File And Function Level Report

File:	OVRLandContainer.sol
Contract:	OVRLandContainer
Observation:	Passed
Test Report:	Passed

Sl.	Function	Type	Observation	Test Report	Conclusion	Score
1	initialize	write	Passed	All Passed	No Issue	Passed
2	ownerOfChild	read	Passed	All Passed	No Issue	Passed
3	childsOfParent	read	Passed	All Passed	No Issue	Passed
4	setMarketplaceAddress	write	Passed	All Passed	No Issue	Passed
5	setRentingAddress	write	Passed	All Passed	No Issue	Passed

6	removeLandFromContainer	write	Passed	All Passed	No Issue	Passed
7	addLandToContainer	write	Passed	All Passed	No Issue	Passed
8	createContainer	write	Passed	All Passed	No Issue	Passed
9	deleteContainer	write	Passed	All Passed	No Issue	Passed
10	addURIEditor	write	Passed	All Passed	No Issue	Passed
11	removeURIEditor	write	Passed	All Passed	No Issue	Passed
12	addUpgrader	write	Passed	All Passed	No Issue	Passed
13	removeUpgrader	write	Passed	All Passed	No Issue	Passed
14	addAdminRole	write	Passed	All Passed	No Issue	Passed
15	removeAdminRole	write	Passed	All Passed	No Issue	Passed
16	setOVRLandContainerURI	write	Passed	All Passed	No Issue	Passed
17	_burn	write	Passed	All Passed	No Issue	Passed
18	tokenURI	read	Passed	All Passed	No Issue	Passed
19	supportsInterface	read	Passed	All Passed	No Issue	Passed

OVRMarketplace

File And Function Level Report

File:	OVRMarketplace.sol
Contract:	OVRMarketplace
Observation:	Passed
Test Report:	Passed

Sl.	Function	Type	Observation	Test Report	Conclusion	Score
1	initialize	write	Passed	All Passed	No Issue	Passed
2	addAdminRole	write	Passed	All Passed	No Issue	Passed
3	removeAdminRole	write	Passed	All Passed	No Issue	Passed
4	lastOffer	read	Passed	All Passed	No Issue	Passed
5	sellView	read	Passed	All Passed	No Issue	Passed
6	landIsOnSelling	read	Passed	All Passed	No Issue	Passed
7	containerIsOnSelling	read	Passed	All Passed	No Issue	Passed
8	sellViewContainer	read	Passed	All Passed	No Issue	Passed
9	setFeeAddr	write	Passed	All Passed	No Issue	Passed
10	setOVRLandContainerAddress	write	Passed	All Passed	No Issue	Passed
11	placeOffer	write	Passed	All Passed	No Issue	Passed
12	acceptOffer	write	Passed	All Passed	No Issue	Passed
13	sell	write	Passed	All Passed	No Issue	Passed
14	sellContainer	write	Passed	All Passed	No Issue	Passed

15	buyContainer	write	Passed	All Passed	No Issue	Passed
16	cancelOffer	write	Passed	All Passed	No Issue	Passed
17	cancelSellContainer	write	Passed	All Passed	No Issue	Passed
18	updatePriceContainer	write	Passed	All Passed	No Issue	Passed
19	updatePriceLand	write	Passed	All Passed	No Issue	Passed
20	cancelSell	write	Passed	All Passed	No Issue	Passed
21	buy	write	Passed	All Passed	No Issue	Passed
22	moneyBack	write	Passed	All Passed	No Issue	Passed

OVRToken

File And Function Level Report

File:	OVRToken.sol
Contract:	OVR
Observation:	Passed
Test Report:	Passed

Sl.	Function	Type	Observation	Test Report	Conclusion	Score
1	transfer	write	Passed	All Passed	No Issue	Passed
2	allowance	read	Passed	All Passed	No Issue	Passed
3	approve	write	Passed	All Passed	No Issue	Passed
4	transferFrom	write	Passed	All Passed	No Issue	Passed

5	increaseAllowance	write	Passed	All Passed	No Issue	Passed
6	decreaseAllowance	write	Passed	All Passed	No Issue	Passed
7	_mint	internal	Passed	All Passed	No Issue	Passed
8	_burn	internal	Passed	All Passed	No Issue	Passed

Uniswapv2router

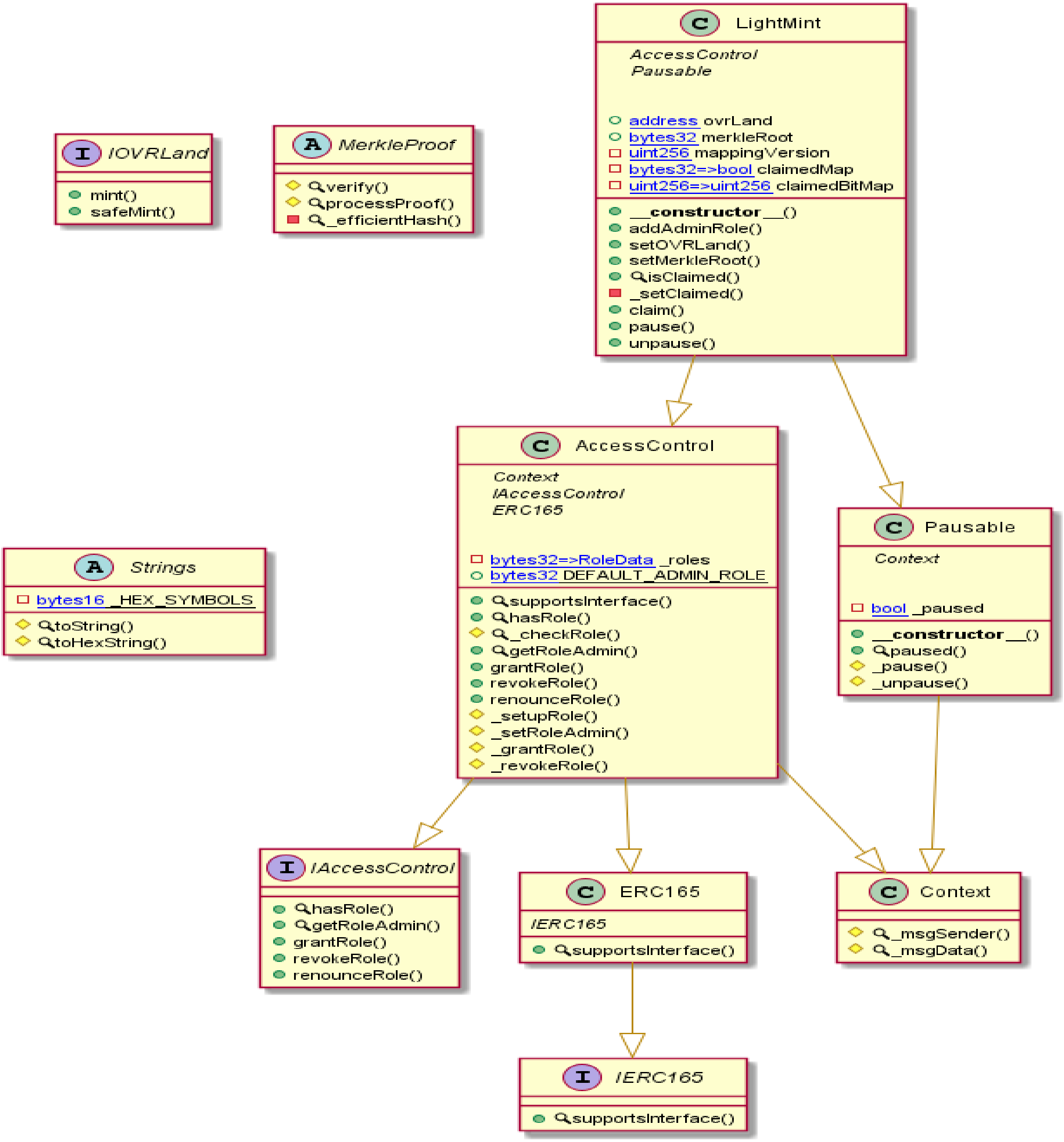
File And Function Level Report

File:	UniswapV2Router.sol
Contract:	UniswapV2Router01
Observation:	Passed
Test Report:	Passed

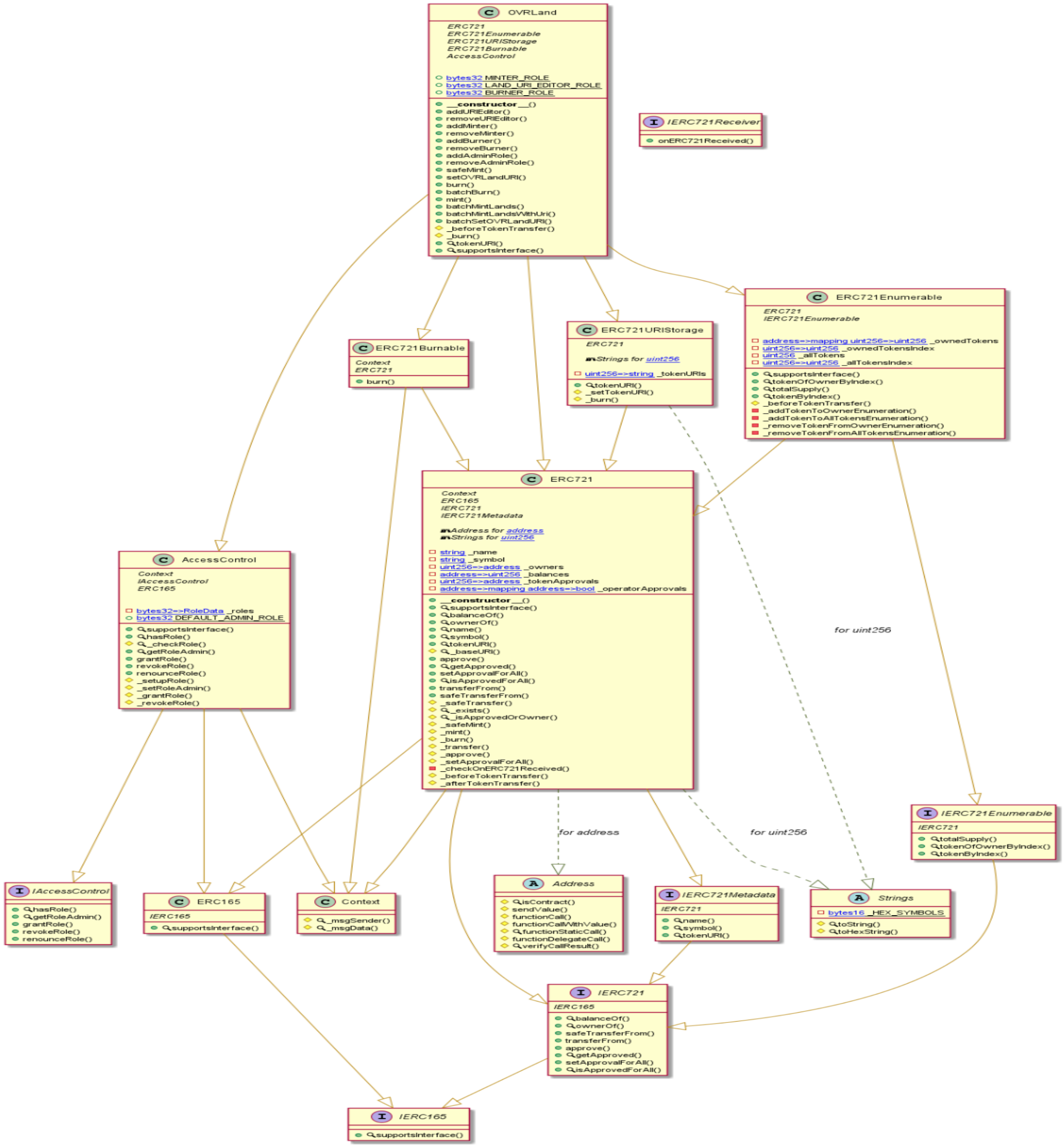
Sl.	Function	Type	Observation	Test Report	Conclusion	Score
1	_addLiquidity	private	Passed	All Passed	No Issue	Passed
2	addLiquidity	write	Passed	All Passed	No Issue	Passed
3	addLiquidityETH	write	Passed	All Passed	No Issue	Passed
4	removeLiquidity	write	Passed	All Passed	No Issue	Passed
5	removeLiquidityETH	write	Passed	All Passed	No Issue	Passed
6	removeLiquidityWithPermit	write	Passed	All Passed	No Issue	Passed

7	removeLiquidityETHWithPermit	write	Passed	All Passed	No Issue	Passed
8	_swap	private	Passed	All Passed	No Issue	Passed
9	swapExactTokensForTokens	write	Passed	All Passed	No Issue	Passed
10	swapTokensForExactTokens	write	Passed	All Passed	No Issue	Passed
11	swapExactETHForTokens	write	Passed	All Passed	No Issue	Passed
12	swapTokensForExactETH	write	Passed	All Passed	No Issue	Passed
13	swapExactTokensForETH	write	Passed	All Passed	No Issue	Passed
14	swapETHForExactTokens	write	Passed	All Passed	No Issue	Passed
15	quote	read	Passed	All Passed	No Issue	Passed
16	getAmountOut	read	Passed	All Passed	No Issue	Passed
17	getAmountIn	read	Passed	All Passed	No Issue	Passed

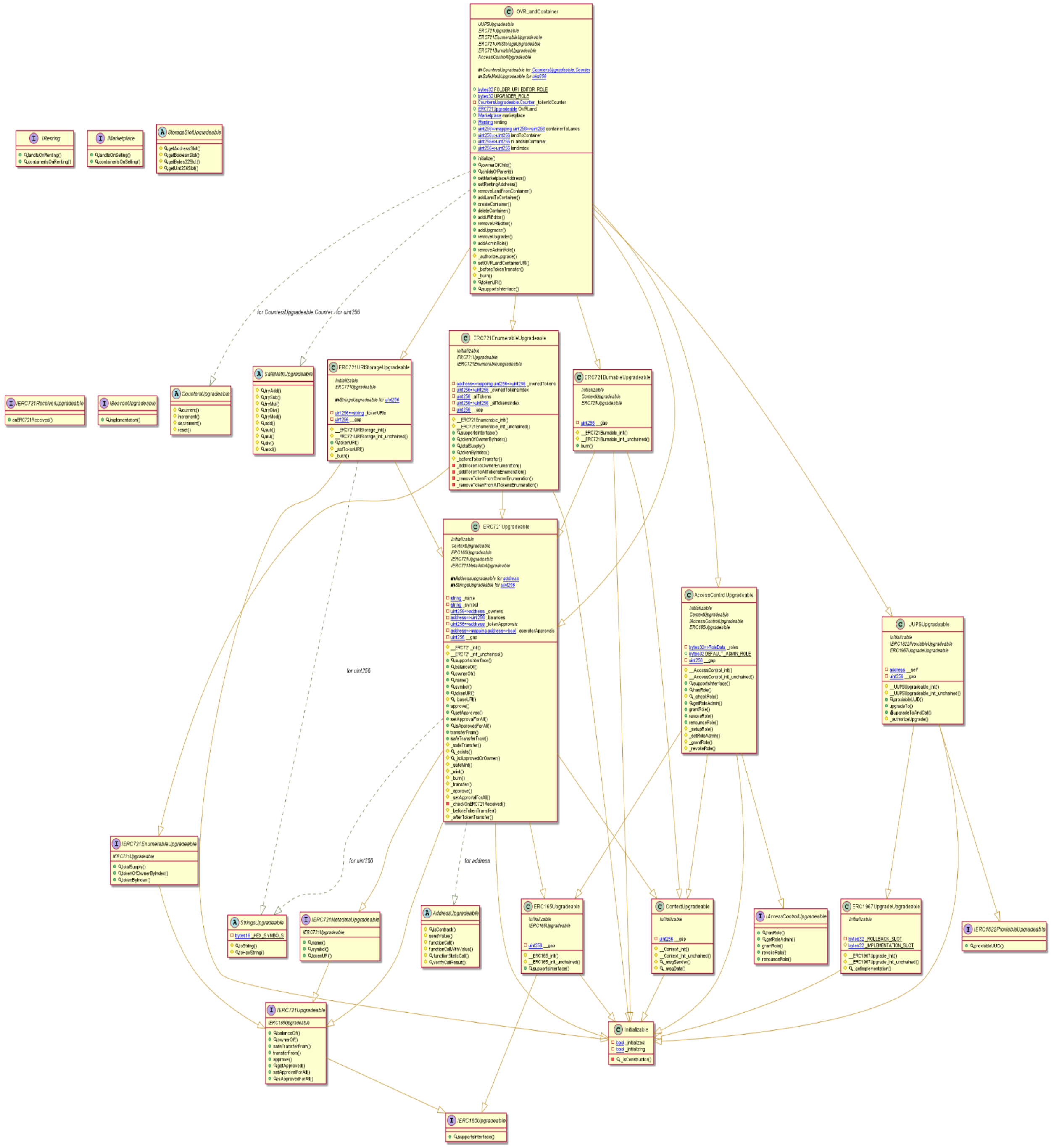
Code Flow Diagram - LightMint



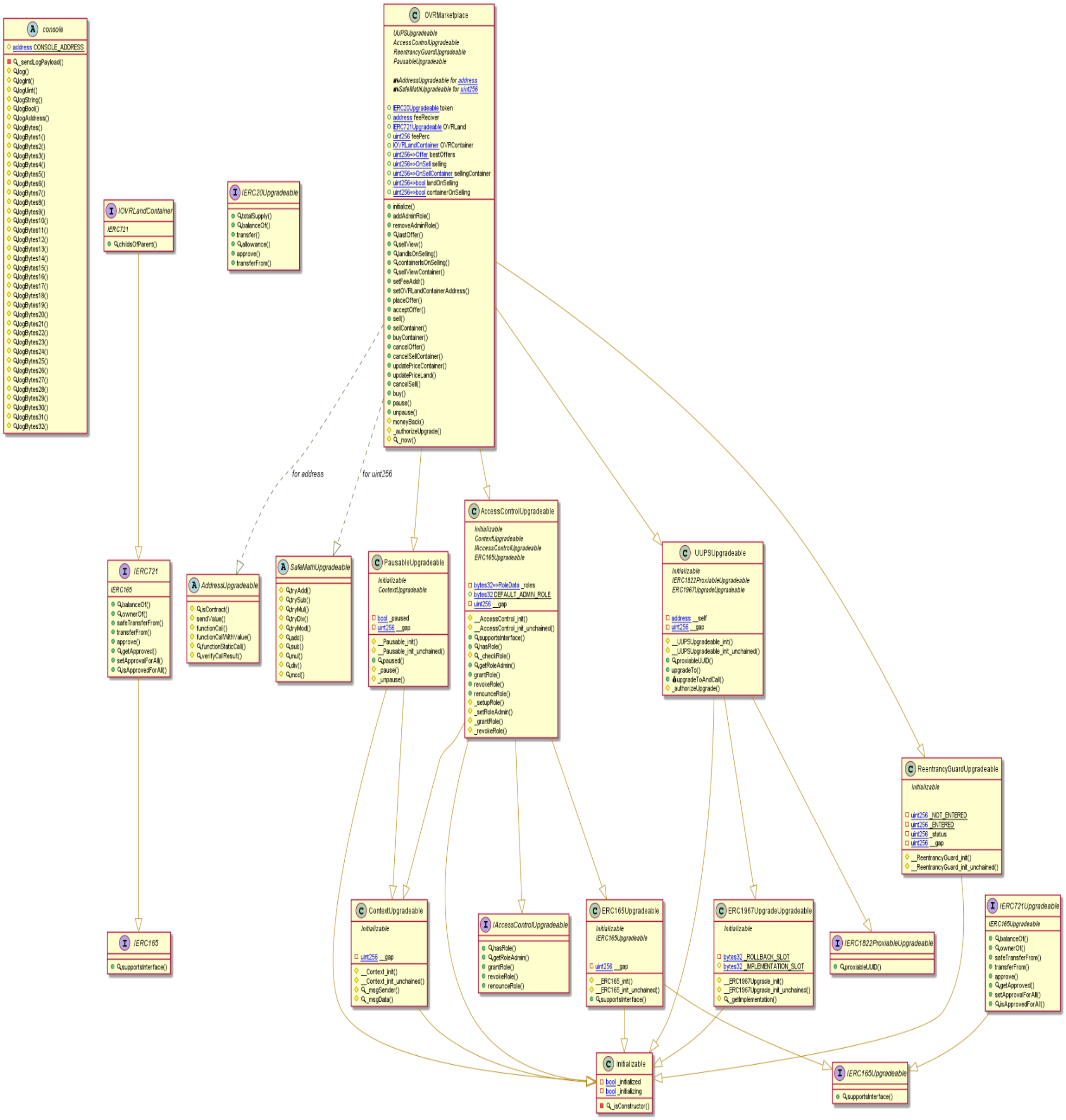
Code Flow Diagram - OVRLand



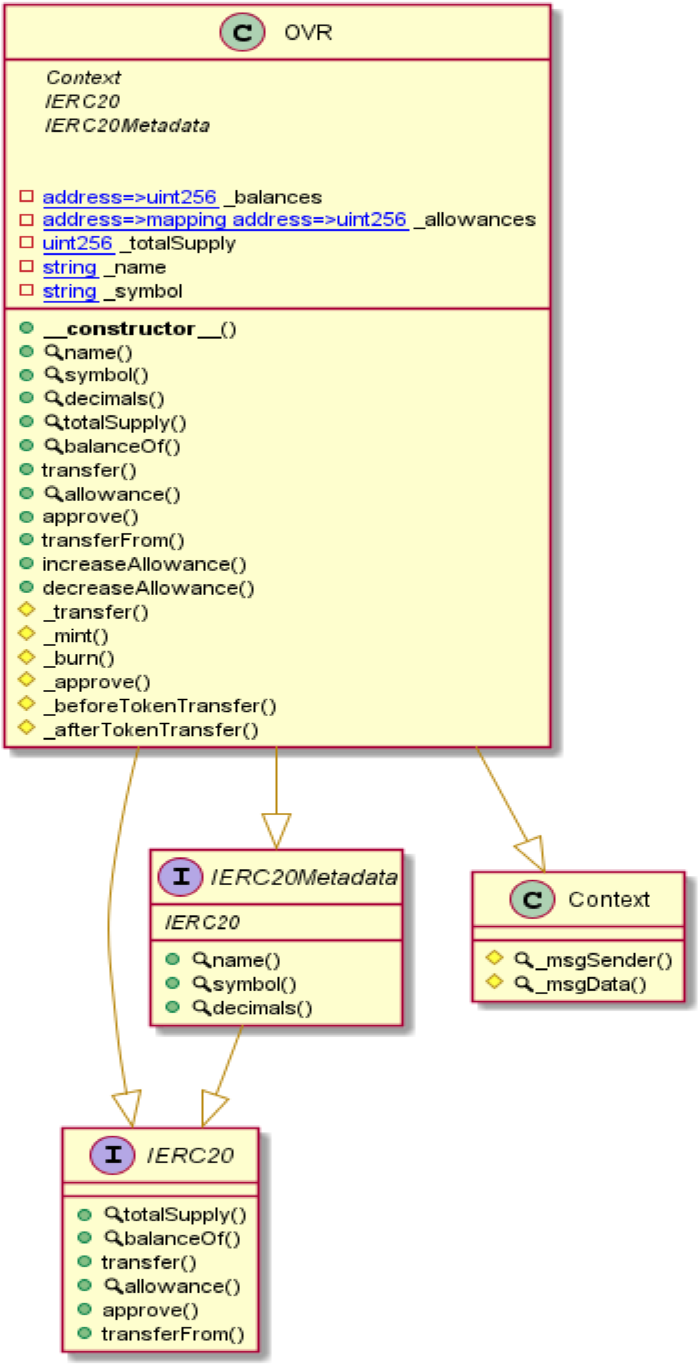
Code Flow Diagram - OVRLandContainer



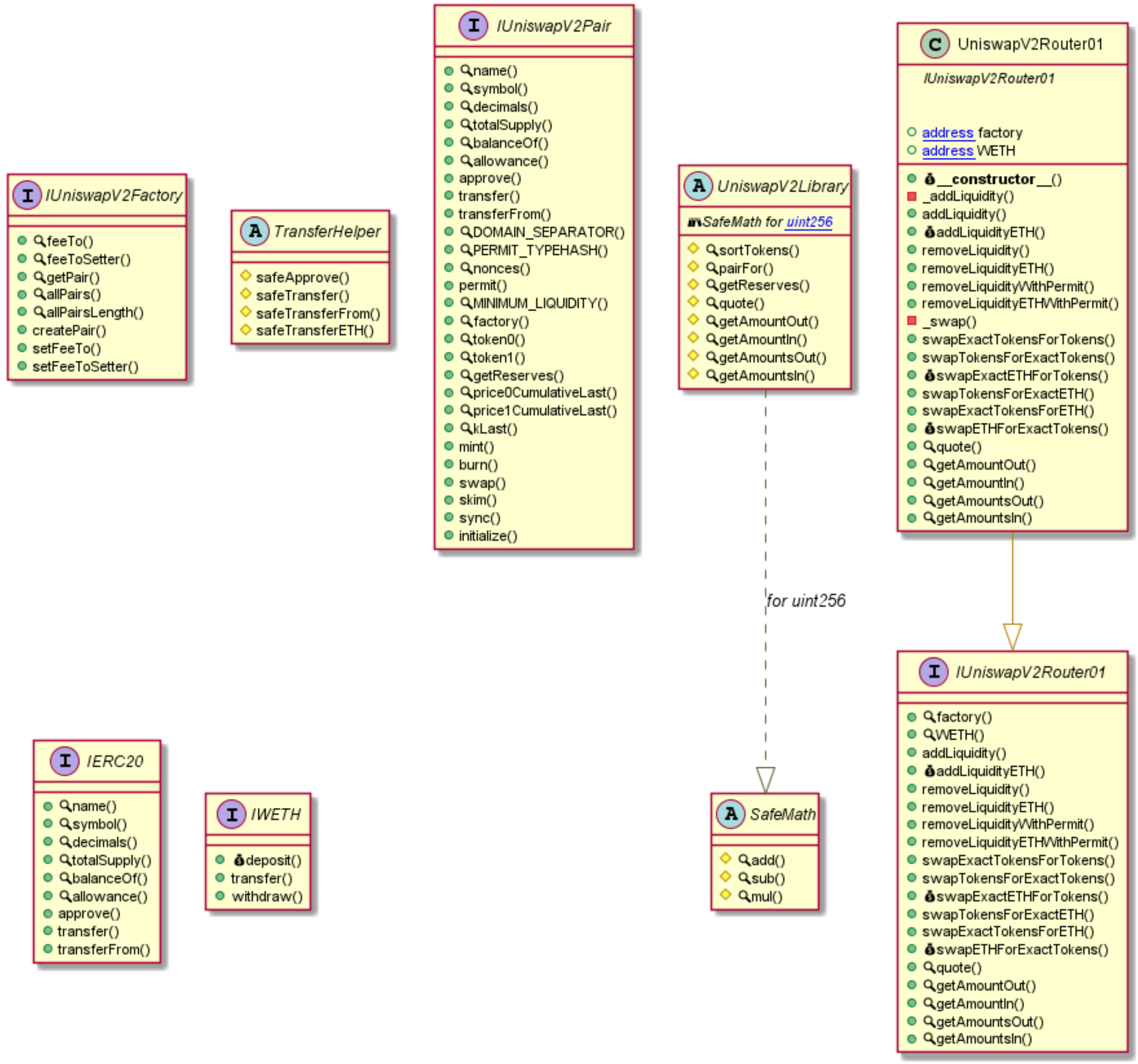
Code Flow Diagram - OVRMarketplace



Code Flow Diagram - OVRToken

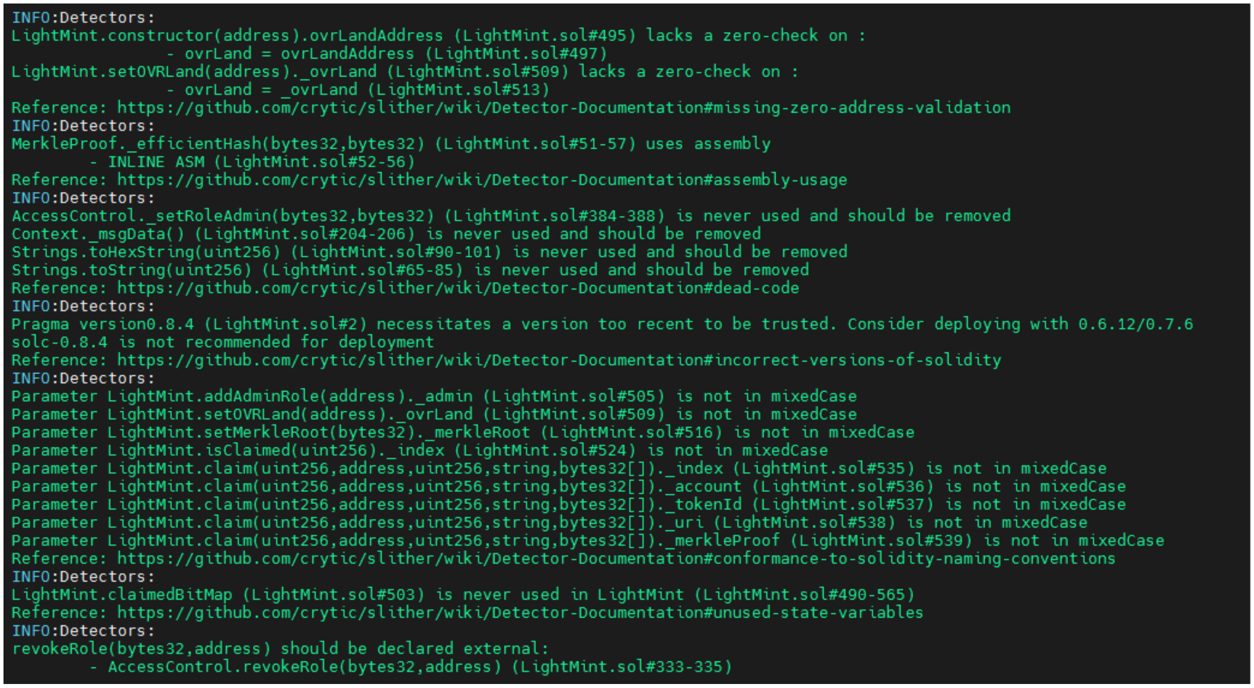


Code Flow Diagram - Uniswapv2router

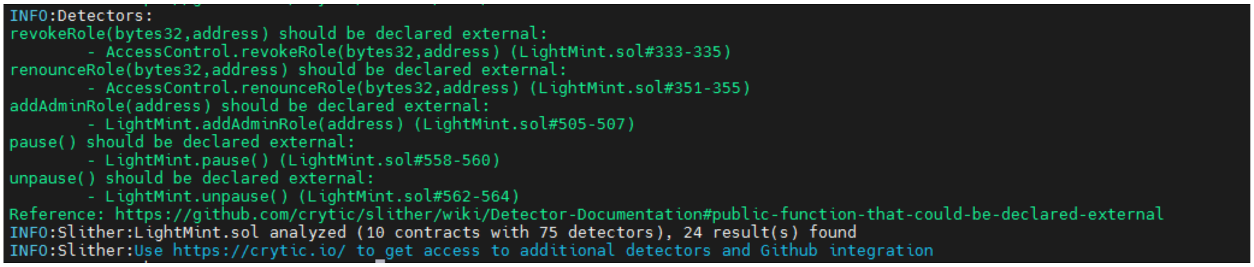


Code Flow Diagram - Slither Results Log

Slither log >> LightMint.sol

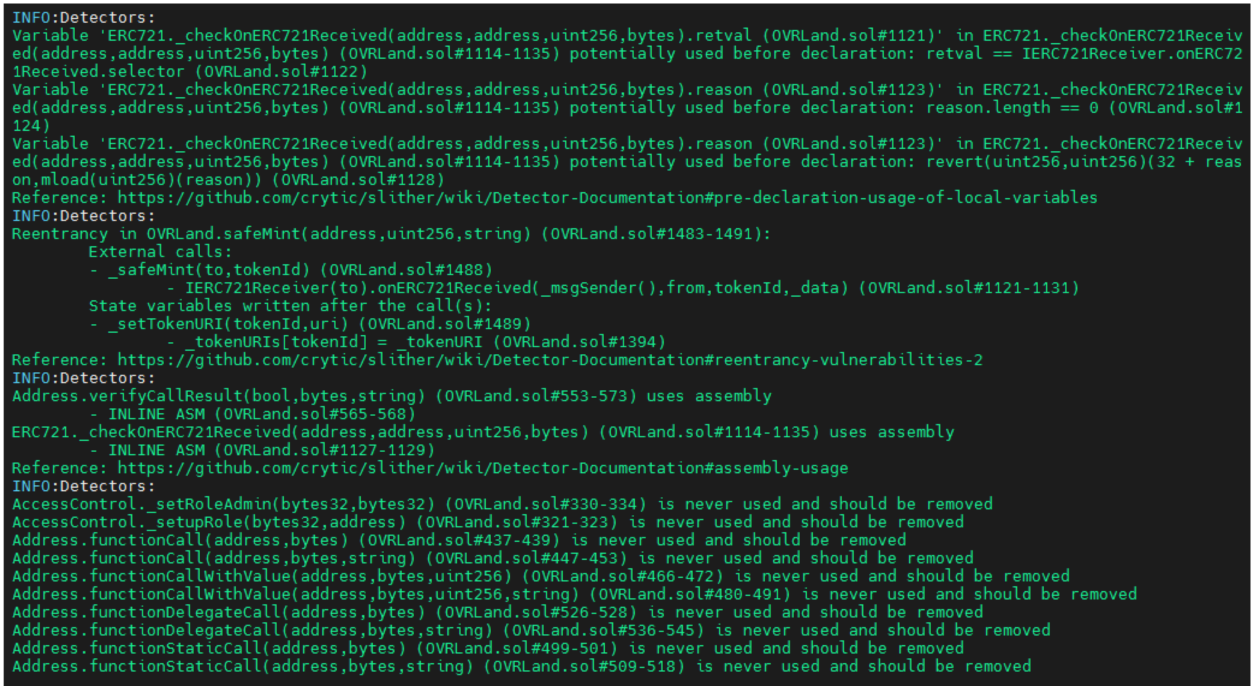


```
INFO:Detectors:
LightMint.constructor(address).ovrLandAddress (LightMint.sol#495) lacks a zero-check on :
                - ovrLand = ovrLandAddress (LightMint.sol#497)
LightMint.setOVRLand(address)._ovrLand (LightMint.sol#509) lacks a zero-check on :
                - ovrLand = _ovrLand (LightMint.sol#513)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#missing-zero-address-validation
INFO:Detectors:
MerkleProof._efficientHash(bytes32,bytes32) (LightMint.sol#51-57) uses assembly
        - INLINE ASM (LightMint.sol#52-56)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#assembly-usage
INFO:Detectors:
AccessControl._setRoleAdmin(bytes32,bytes32) (LightMint.sol#384-388) is never used and should be removed
Context._msgData() (LightMint.sol#204-206) is never used and should be removed
Strings.toHexString(uint256) (LightMint.sol#90-101) is never used and should be removed
Strings.toString(uint256) (LightMint.sol#65-85) is never used and should be removed
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dead-code
INFO:Detectors:
Pragma version0.8.4 (LightMint.sol#2) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
solc-0.8.4 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-versions-of-solidity
INFO:Detectors:
Parameter LightMint.addAdminRole(address)._admin (LightMint.sol#505) is not in mixedCase
Parameter LightMint.setOVRLand(address)._ovrLand (LightMint.sol#509) is not in mixedCase
Parameter LightMint.setMerkleRoot(bytes32)._merkleRoot (LightMint.sol#516) is not in mixedCase
Parameter LightMint.isClaimed(uint256)._index (LightMint.sol#524) is not in mixedCase
Parameter LightMint.claim(uint256,address,uint256,string,bytes32[])._index (LightMint.sol#535) is not in mixedCase
Parameter LightMint.claim(uint256,address,uint256,string,bytes32[])._account (LightMint.sol#536) is not in mixedCase
Parameter LightMint.claim(uint256,address,uint256,string,bytes32[])._tokenId (LightMint.sol#537) is not in mixedCase
Parameter LightMint.claim(uint256,address,uint256,string,bytes32[])._uri (LightMint.sol#538) is not in mixedCase
Parameter LightMint.claim(uint256,address,uint256,string,bytes32[])._merkleProof (LightMint.sol#539) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-solidity-naming-conventions
INFO:Detectors:
LightMint.claimedBitMap (LightMint.sol#503) is never used in LightMint (LightMint.sol#490-565)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#unused-state-variables
INFO:Detectors:
revokeRole(bytes32,address) should be declared external:
        - AccessControl.revokeRole(bytes32,address) (LightMint.sol#333-335)
```

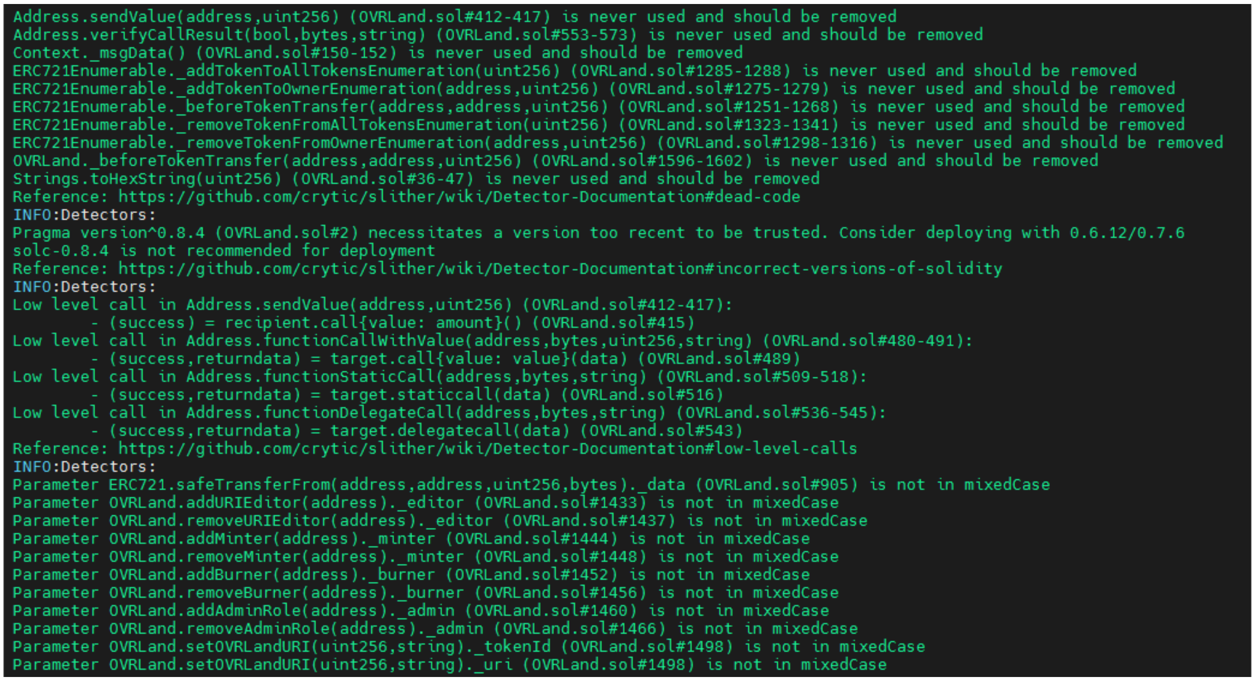


```
INFO:Detectors:
revokeRole(bytes32,address) should be declared external:
        - AccessControl.revokeRole(bytes32,address) (LightMint.sol#333-335)
renounceRole(bytes32,address) should be declared external:
        - AccessControl.renounceRole(bytes32,address) (LightMint.sol#351-355)
addAdminRole(address) should be declared external:
        - LightMint.addAdminRole(address) (LightMint.sol#505-507)
pause() should be declared external:
        - LightMint.pause() (LightMint.sol#558-560)
unpause() should be declared external:
        - LightMint.unpause() (LightMint.sol#562-564)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#public-function-that-could-be-declared-external
INFO:Slither:LightMint.sol analyzed (10 contracts with 75 detectors), 24 result(s) found
INFO:Slither:Use https://crytic.io/ to get access to additional detectors and Github integration
```

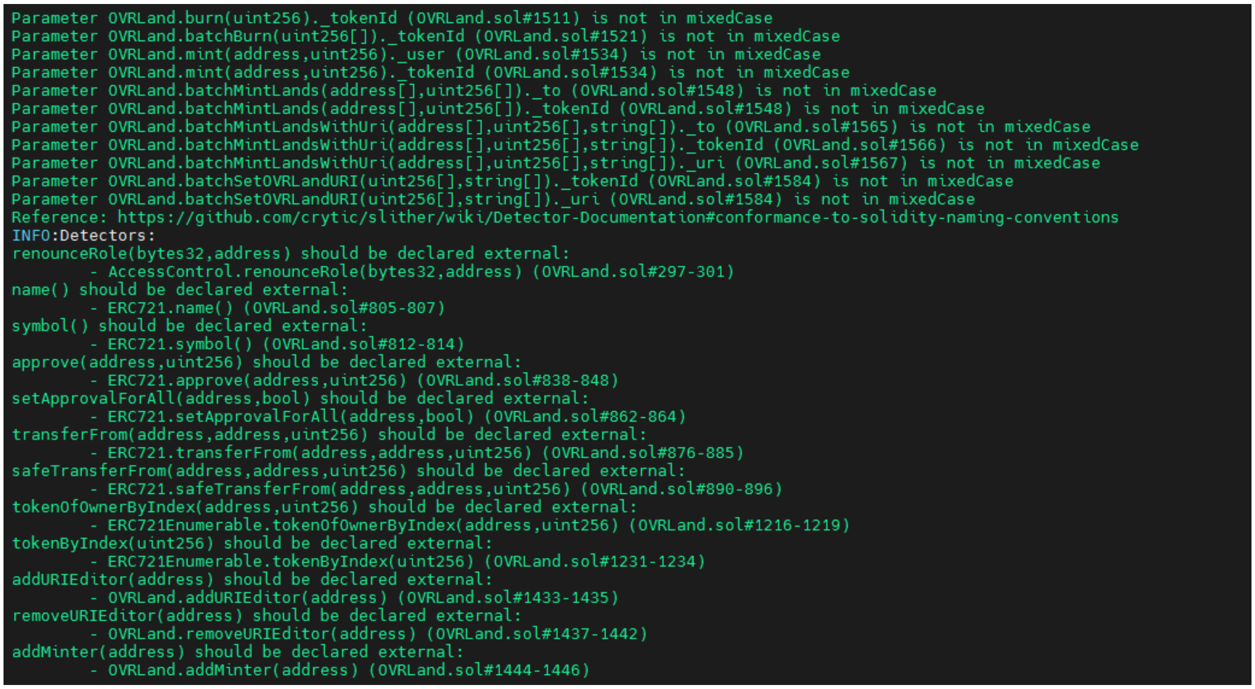
Slither log >> OVRLand.sol



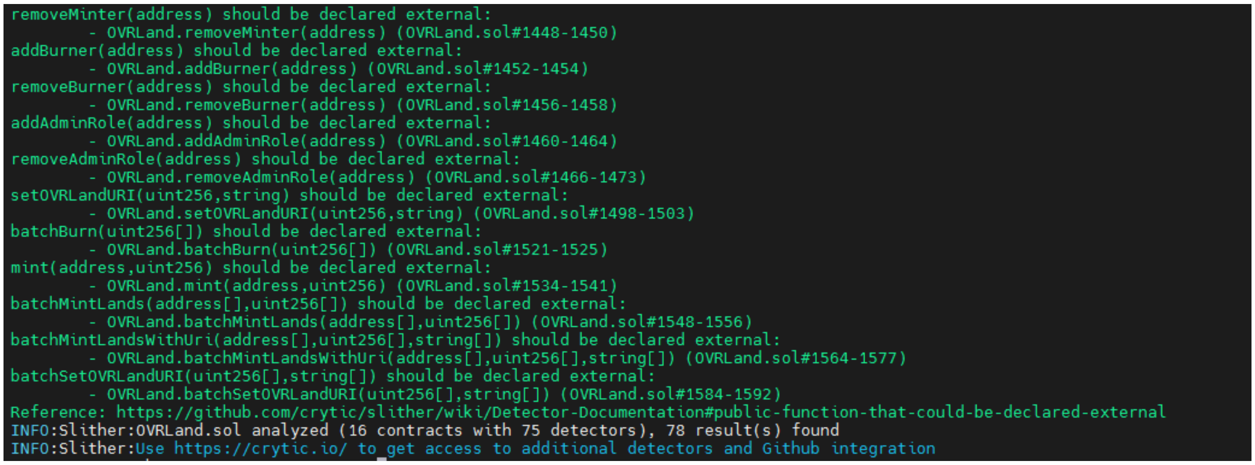
```
INFO:Detectors:
Variable 'ERC721._checkOnERC721Received(address,address,uint256,bytes).retval (OVRLand.sol#1121)' in ERC721._checkOnERC721Receiv
ed(address,address,uint256,bytes) (OVRLand.sol#1114-1135) potentially used before declaration: retval == IERC721Receiver.onERC72
1Received.selector (OVRLand.sol#1122)
Variable 'ERC721._checkOnERC721Received(address,address,uint256,bytes).reason (OVRLand.sol#1123)' in ERC721._checkOnERC721Receiv
ed(address,address,uint256,bytes) (OVRLand.sol#1114-1135) potentially used before declaration: reason.length == 0 (OVRLand.sol#1
124)
Variable 'ERC721._checkOnERC721Received(address,address,uint256,bytes).reason (OVRLand.sol#1123)' in ERC721._checkOnERC721Receiv
ed(address,address,uint256,bytes) (OVRLand.sol#1114-1135) potentially used before declaration: revert(uint256,uint256)(32 + reas
on,mload(uint256)(reason)) (OVRLand.sol#1128)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#pre-declaration-usage-of-local-variables
INFO:Detectors:
Reentrancy in OVRLand.safeMint(address,uint256,string) (OVRLand.sol#1483-1491):
        External calls:
        - _safeMint(to,tokenId) (OVRLand.sol#1488)
                - IERC721Receiver(to).onERC721Received(_msgSender(),from,tokenId,_data) (OVRLand.sol#1121-1131)
        State variables written after the call(s):
        - _setTokenURI(tokenId,uri) (OVRLand.sol#1489)
                - _tokenURIs[tokenId] = _tokenURI (OVRLand.sol#1394)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-vulnerabilities-2
INFO:Detectors:
Address.verifyCallResult(bool,bytes,string) (OVRLand.sol#553-573) uses assembly
        - INLINE ASM (OVRLand.sol#565-568)
ERC721._checkOnERC721Received(address,address,uint256,bytes) (OVRLand.sol#1114-1135) uses assembly
        - INLINE ASM (OVRLand.sol#1127-1129)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#assembly-usage
INFO:Detectors:
AccessControl._setRoleAdmin(bytes32,bytes32) (OVRLand.sol#330-334) is never used and should be removed
AccessControl._setupRole(bytes32,address) (OVRLand.sol#321-323) is never used and should be removed
Address.functionCall(address,bytes) (OVRLand.sol#437-439) is never used and should be removed
Address.functionCall(address,bytes,string) (OVRLand.sol#447-453) is never used and should be removed
Address.functionCallWithValue(address,bytes,uint256) (OVRLand.sol#466-472) is never used and should be removed
Address.functionCallWithValue(address,bytes,uint256,string) (OVRLand.sol#480-491) is never used and should be removed
Address.functionDelegateCall(address,bytes) (OVRLand.sol#526-528) is never used and should be removed
Address.functionDelegateCall(address,bytes,string) (OVRLand.sol#536-545) is never used and should be removed
Address.functionStaticCall(address,bytes) (OVRLand.sol#499-501) is never used and should be removed
Address.functionStaticCall(address,bytes,string) (OVRLand.sol#509-518) is never used and should be removed
```



```
Address.sendValue(address,uint256) (OVRLand.sol#412-417) is never used and should be removed
Address.verifyCallResult(bool,bytes,string) (OVRLand.sol#553-573) is never used and should be removed
Context._msgData() (OVRLand.sol#150-152) is never used and should be removed
ERC721Enumerable._addTokenToAllTokensEnumeration(uint256) (OVRLand.sol#1285-1288) is never used and should be removed
ERC721Enumerable._addTokenToOwnerEnumeration(address,uint256) (OVRLand.sol#1275-1279) is never used and should be removed
ERC721Enumerable._beforeTokenTransfer(address,address,uint256) (OVRLand.sol#1251-1268) is never used and should be removed
ERC721Enumerable._removeTokenFromAllTokensEnumeration(uint256) (OVRLand.sol#1323-1341) is never used and should be removed
ERC721Enumerable._removeTokenFromOwnerEnumeration(address,uint256) (OVRLand.sol#1298-1316) is never used and should be removed
OVRLand._beforeTokenTransfer(address,address,uint256) (OVRLand.sol#1596-1602) is never used and should be removed
Strings.toHexString(uint256) (OVRLand.sol#36-47) is never used and should be removed
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dead-code
INFO:Detectors:
Pragma version^0.8.4 (OVRLand.sol#2) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
solc-0.8.4 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-versions-of-solidity
INFO:Detectors:
Low level call in Address.sendValue(address,uint256) (OVRLand.sol#412-417):
        - (success) = recipient.call{value: amount}() (OVRLand.sol#415)
Low level call in Address.functionCallWithValue(address,bytes,uint256,string) (OVRLand.sol#480-491):
        - (success,returndata) = target.call{value: value}(data) (OVRLand.sol#489)
Low level call in Address.functionStaticCall(address,bytes,string) (OVRLand.sol#509-518):
        - (success,returndata) = target.staticcall(data) (OVRLand.sol#516)
Low level call in Address.functionDelegateCall(address,bytes,string) (OVRLand.sol#536-545):
        - (success,returndata) = target.delegatecall(data) (OVRLand.sol#543)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#low-level-calls
INFO:Detectors:
Parameter ERC721.safeTransferFrom(address,address,uint256,bytes)._data (OVRLand.sol#905) is not in mixedCase
Parameter OVRLand.addURIEditor(address)._editor (OVRLand.sol#1433) is not in mixedCase
Parameter OVRLand.removeURIEditor(address)._editor (OVRLand.sol#1437) is not in mixedCase
Parameter OVRLand.addMinter(address)._minter (OVRLand.sol#1444) is not in mixedCase
Parameter OVRLand.removeMinter(address)._minter (OVRLand.sol#1448) is not in mixedCase
Parameter OVRLand.addBurner(address)._burner (OVRLand.sol#1452) is not in mixedCase
Parameter OVRLand.removeBurner(address)._burner (OVRLand.sol#1456) is not in mixedCase
Parameter OVRLand.addAdminRole(address)._admin (OVRLand.sol#1460) is not in mixedCase
Parameter OVRLand.removeAdminRole(address)._admin (OVRLand.sol#1466) is not in mixedCase
Parameter OVRLand.setOVRLandURI(uint256,string)._tokenId (OVRLand.sol#1498) is not in mixedCase
Parameter OVRLand.setOVRLandURI(uint256,string)._uri (OVRLand.sol#1498) is not in mixedCase
```

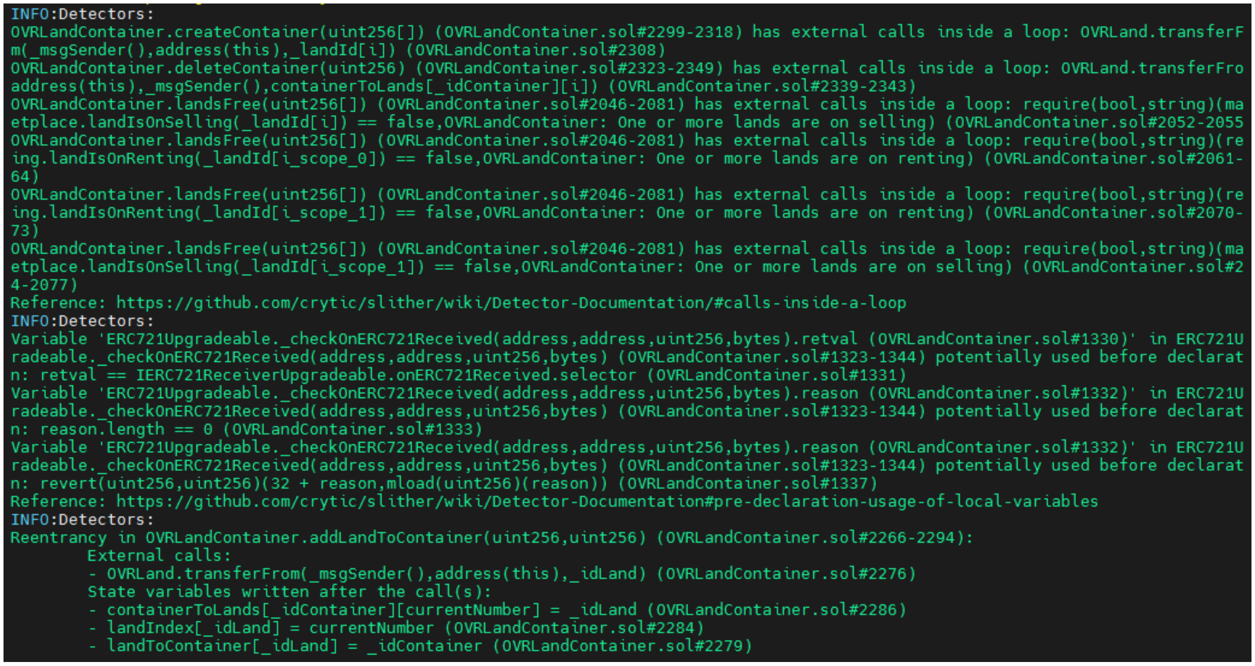


```
Parameter OVRLand.burn(uint256)._tokenId (OVRLand.sol#1511) is not in mixedCase
Parameter OVRLand.batchBurn(uint256[])._tokenId (OVRLand.sol#1521) is not in mixedCase
Parameter OVRLand.mint(address,uint256)._user (OVRLand.sol#1534) is not in mixedCase
Parameter OVRLand.mint(address,uint256)._tokenId (OVRLand.sol#1534) is not in mixedCase
Parameter OVRLand.batchMintLands(address[],uint256[])._to (OVRLand.sol#1548) is not in mixedCase
Parameter OVRLand.batchMintLands(address[],uint256[])._tokenId (OVRLand.sol#1548) is not in mixedCase
Parameter OVRLand.batchMintLandsWithUri(address[],uint256[],string[])._to (OVRLand.sol#1565) is not in mixedCase
Parameter OVRLand.batchMintLandsWithUri(address[],uint256[],string[])._tokenId (OVRLand.sol#1566) is not in mixedCase
Parameter OVRLand.batchMintLandsWithUri(address[],uint256[],string[])._uri (OVRLand.sol#1567) is not in mixedCase
Parameter OVRLand.batchSetOVRLandURI(uint256[],string[])._tokenId (OVRLand.sol#1584) is not in mixedCase
Parameter OVRLand.batchSetOVRLandURI(uint256[],string[])._uri (OVRLand.sol#1584) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-solidity-naming-conventions
INFO:Detectors:
renounceRole(bytes32,address) should be declared external:
        - AccessControl.renounceRole(bytes32,address) (OVRLand.sol#297-301)
name() should be declared external:
        - ERC721.name() (OVRLand.sol#805-807)
symbol() should be declared external:
        - ERC721.symbol() (OVRLand.sol#812-814)
approve(address,uint256) should be declared external:
        - ERC721.approve(address,uint256) (OVRLand.sol#838-848)
setApprovalForAll(address,bool) should be declared external:
        - ERC721.setApprovalForAll(address,bool) (OVRLand.sol#862-864)
transferFrom(address,address,uint256) should be declared external:
        - ERC721.transferFrom(address,address,uint256) (OVRLand.sol#876-885)
safeTransferFrom(address,address,uint256) should be declared external:
        - ERC721.safeTransferFrom(address,address,uint256) (OVRLand.sol#890-896)
tokenOfOwnerByIndex(address,uint256) should be declared external:
        - ERC721Enumerable.tokenOfOwnerByIndex(address,uint256) (OVRLand.sol#1216-1219)
tokenByIndex(uint256) should be declared external:
        - ERC721Enumerable.tokenByIndex(uint256) (OVRLand.sol#1231-1234)
addURIEditor(address) should be declared external:
        - OVRLand.addURIEditor(address) (OVRLand.sol#1433-1435)
removeURIEditor(address) should be declared external:
        - OVRLand.removeURIEditor(address) (OVRLand.sol#1437-1442)
addMinter(address) should be declared external:
        - OVRLand.addMinter(address) (OVRLand.sol#1444-1446)
```

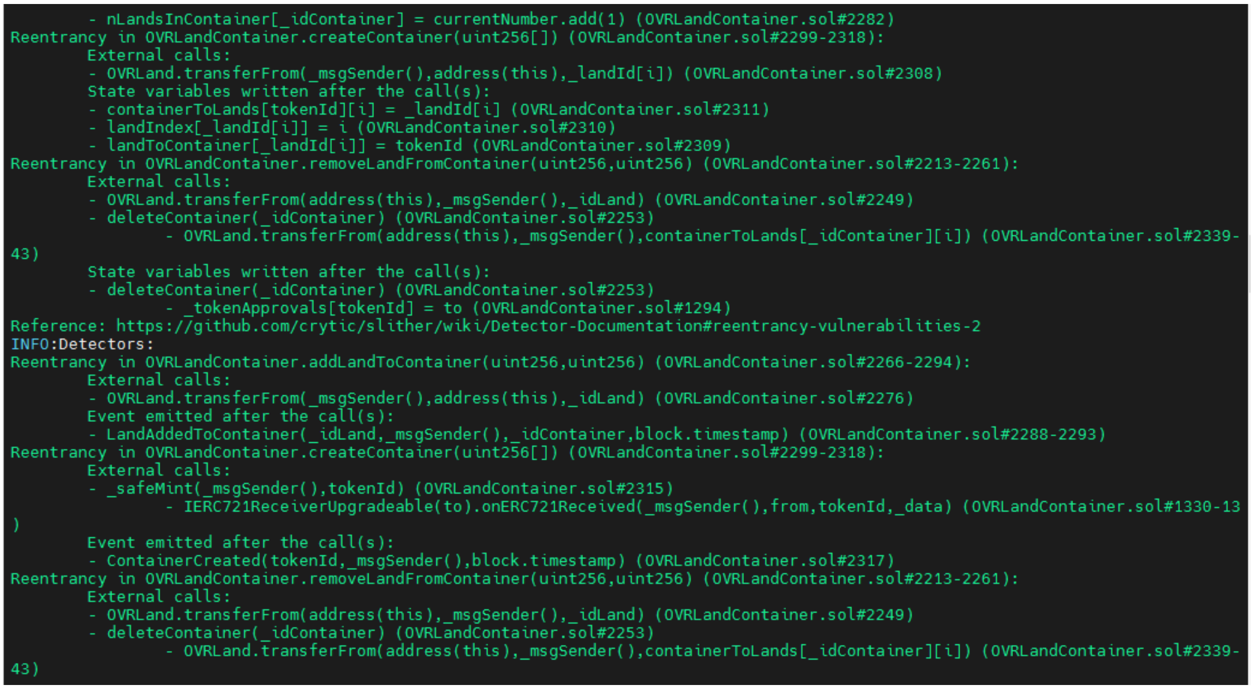


```
removeMinter(address) should be declared external:
        - OVRLand.removeMinter(address) (OVRLand.sol#1448-1450)
addBurner(address) should be declared external:
        - OVRLand.addBurner(address) (OVRLand.sol#1452-1454)
removeBurner(address) should be declared external:
        - OVRLand.removeBurner(address) (OVRLand.sol#1456-1458)
addAdminRole(address) should be declared external:
        - OVRLand.addAdminRole(address) (OVRLand.sol#1460-1464)
removeAdminRole(address) should be declared external:
        - OVRLand.removeAdminRole(address) (OVRLand.sol#1466-1473)
setOVRLandURI(uint256,string) should be declared external:
        - OVRLand.setOVRLandURI(uint256,string) (OVRLand.sol#1498-1503)
batchBurn(uint256[]) should be declared external:
        - OVRLand.batchBurn(uint256[]) (OVRLand.sol#1521-1525)
mint(address,uint256) should be declared external:
        - OVRLand.mint(address,uint256) (OVRLand.sol#1534-1541)
batchMintLands(address[],uint256[]) should be declared external:
        - OVRLand.batchMintLands(address[],uint256[]) (OVRLand.sol#1548-1556)
batchMintLandsWithUri(address[],uint256[],string[]) should be declared external:
        - OVRLand.batchMintLandsWithUri(address[],uint256[],string[]) (OVRLand.sol#1564-1577)
batchSetOVRLandURI(uint256[],string[]) should be declared external:
        - OVRLand.batchSetOVRLandURI(uint256[],string[]) (OVRLand.sol#1584-1592)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#public-function-that-could-be-declared-external
INFO:Slither:OVRLand.sol analyzed (16 contracts with 75 detectors), 78 result(s) found
INFO:Slither:Use https://crytic.io/ to get access to additional detectors and Github integration
```

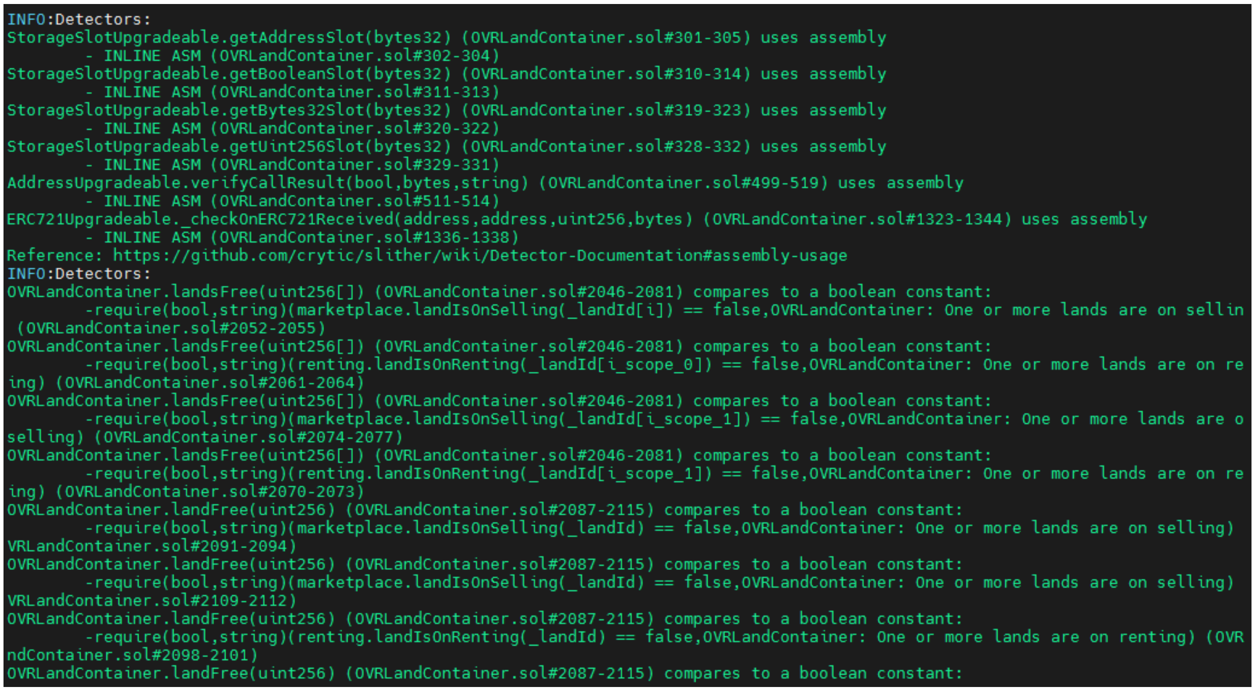
Slither log >> OVRLandContainer.sol



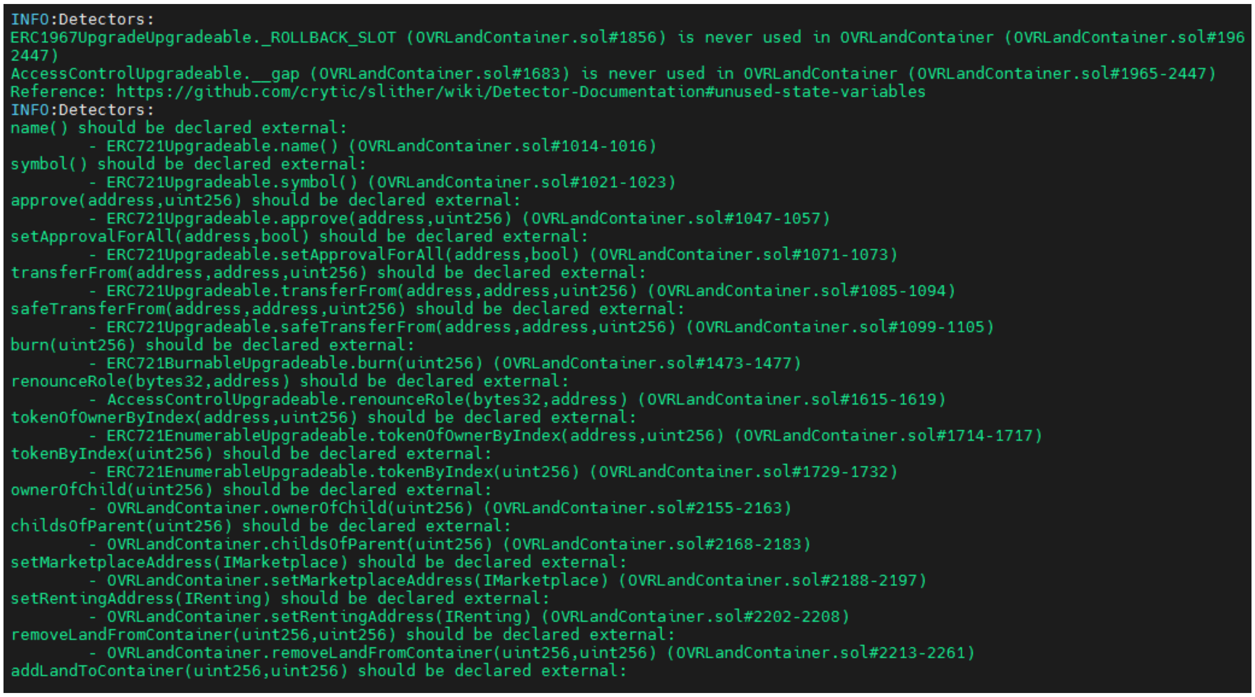
```
INFO:Detectors:
OVRLandContainer.createContainer(uint256[]) (OVRLandContainer.sol#2299-2318) has external calls inside a loop: OVRLand.transferF
m(_msgSender(),address(this),_landId[i]) (OVRLandContainer.sol#2308)
OVRLandContainer.deleteContainer(uint256) (OVRLandContainer.sol#2323-2349) has external calls inside a loop: OVRLand.transferFro
address(this),_msgSender(),containerToLands[_idContainer][i]) (OVRLandContainer.sol#2339-2343)
OVRLandContainer.landsFree(uint256[]) (OVRLandContainer.sol#2046-2081) has external calls inside a loop: require(bool,string)(ma
etplace.landIsOnSelling(_landId[i]) == false,OVRLandContainer: One or more lands are on selling) (OVRLandContainer.sol#2052-2055
OVRLandContainer.landsFree(uint256[]) (OVRLandContainer.sol#2046-2081) has external calls inside a loop: require(bool,string)(re
ing.landIsOnRenting(_landId[i_scope_0]) == false,OVRLandContainer: One or more lands are on renting) (OVRLandContainer.sol#2061-
64)
OVRLandContainer.landsFree(uint256[]) (OVRLandContainer.sol#2046-2081) has external calls inside a loop: require(bool,string)(re
ing.landIsOnRenting(_landId[i_scope_1]) == false,OVRLandContainer: One or more lands are on renting) (OVRLandContainer.sol#2070-
73)
OVRLandContainer.landsFree(uint256[]) (OVRLandContainer.sol#2046-2081) has external calls inside a loop: require(bool,string)(ma
etplace.landIsOnSelling(_landId[i_scope_1]) == false,OVRLandContainer: One or more lands are on selling) (OVRLandContainer.sol#2
4-2077)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation/#calls-inside-a-loop
INFO:Detectors:
Variable 'ERC721Upgradeable._checkOnERC721Received(address,address,uint256,bytes).retval (OVRLandContainer.sol#1330)' in ERC721U
radeable._checkOnERC721Received(address,address,uint256,bytes) (OVRLandContainer.sol#1323-1344) potentially used before declarat
n: retval == IERC721ReceiverUpgradeable.onERC721Received.selector (OVRLandContainer.sol#1331)
Variable 'ERC721Upgradeable._checkOnERC721Received(address,address,uint256,bytes).reason (OVRLandContainer.sol#1332)' in ERC721U
radeable._checkOnERC721Received(address,address,uint256,bytes) (OVRLandContainer.sol#1323-1344) potentially used before declarat
n: reason.length == 0 (OVRLandContainer.sol#1333)
Variable 'ERC721Upgradeable._checkOnERC721Received(address,address,uint256,bytes).reason (OVRLandContainer.sol#1332)' in ERC721U
radeable._checkOnERC721Received(address,address,uint256,bytes) (OVRLandContainer.sol#1323-1344) potentially used before declarat
n: revert(uint256,uint256)(32 + reason,mload(uint256)(reason)) (OVRLandContainer.sol#1337)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#pre-declaration-usage-of-local-variables
INFO:Detectors:
Reentrancy in OVRLandContainer.addLandToContainer(uint256,uint256) (OVRLandContainer.sol#2266-2294):
        External calls:
        - OVRLand.transferFrom(_msgSender(),address(this),_idLand) (OVRLandContainer.sol#2276)
        State variables written after the call(s):
        - containerToLands[_idContainer][currentNumber] = _idLand (OVRLandContainer.sol#2286)
        - landIndex[_idLand] = currentNumber (OVRLandContainer.sol#2284)
        - landToContainer[_idLand] = _idContainer (OVRLandContainer.sol#2279)
```



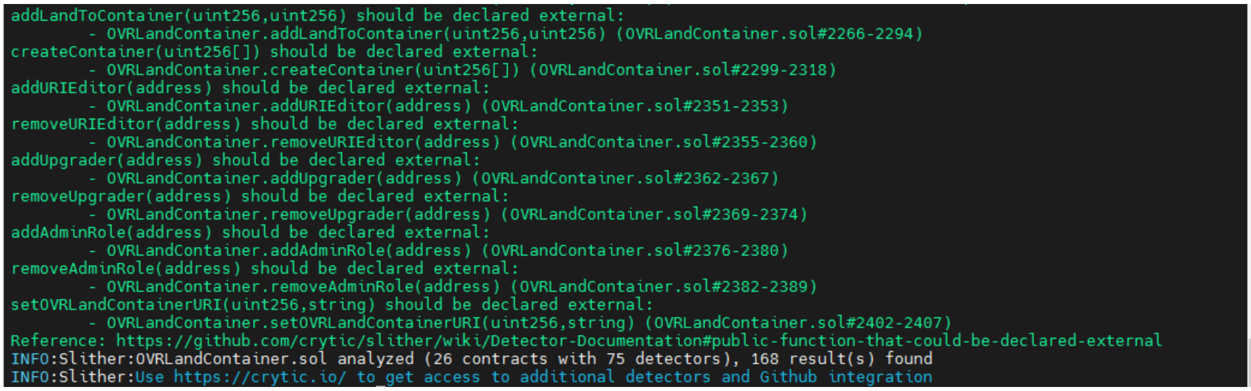
```
        - nLandsInContainer[_idContainer] = currentNumber.add(1) (OVRLandContainer.sol#2282)
Reentrancy in OVRLandContainer.createContainer(uint256[]) (OVRLandContainer.sol#2299-2318):
        External calls:
        - OVRLand.transferFrom(_msgSender(),address(this),_landId[i]) (OVRLandContainer.sol#2308)
        State variables written after the call(s):
        - containerToLands[tokenId][i] = _landId[i] (OVRLandContainer.sol#2311)
        - landIndex[_landId[i]] = i (OVRLandContainer.sol#2310)
        - landToContainer[_landId[i]] = tokenId (OVRLandContainer.sol#2309)
Reentrancy in OVRLandContainer.removeLandFromContainer(uint256,uint256) (OVRLandContainer.sol#2213-2261):
        External calls:
        - OVRLand.transferFrom(address(this),_msgSender(),_idLand) (OVRLandContainer.sol#2249)
        - deleteContainer(_idContainer) (OVRLandContainer.sol#2253)
                - OVRLand.transferFrom(address(this),_msgSender(),containerToLands[_idContainer][i]) (OVRLandContainer.sol#2339-
43)
        State variables written after the call(s):
        - deleteContainer(_idContainer) (OVRLandContainer.sol#2253)
                - _tokenApprovals[tokenId] = to (OVRLandContainer.sol#1294)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-vulnerabilities-2
INFO:Detectors:
Reentrancy in OVRLandContainer.addLandToContainer(uint256,uint256) (OVRLandContainer.sol#2266-2294):
        External calls:
        - OVRLand.transferFrom(_msgSender(),address(this),_idLand) (OVRLandContainer.sol#2276)
        Event emitted after the call(s):
        - LandAddedToContainer(_idLand,_msgSender(),_idContainer,block.timestamp) (OVRLandContainer.sol#2288-2293)
Reentrancy in OVRLandContainer.createContainer(uint256[]) (OVRLandContainer.sol#2299-2318):
        External calls:
        - _safeMint(_msgSender(),tokenId) (OVRLandContainer.sol#2315)
                - IERC721ReceiverUpgradeable(to).onERC721Received(_msgSender(),from,tokenId,_data) (OVRLandContainer.sol#1330-13
)
        Event emitted after the call(s):
        - ContainerCreated(tokenId,_msgSender(),block.timestamp) (OVRLandContainer.sol#2317)
Reentrancy in OVRLandContainer.removeLandFromContainer(uint256,uint256) (OVRLandContainer.sol#2213-2261):
        External calls:
        - OVRLand.transferFrom(address(this),_msgSender(),_idLand) (OVRLandContainer.sol#2249)
        - deleteContainer(_idContainer) (OVRLandContainer.sol#2253)
                - OVRLand.transferFrom(address(this),_msgSender(),containerToLands[_idContainer][i]) (OVRLandContainer.sol#2339-
43)
```



```
INFO:Detectors:
StorageSlotUpgradeable.getAddressSlot(bytes32) (OVRLandContainer.sol#301-305) uses assembly
        - INLINE ASM (OVRLandContainer.sol#302-304)
StorageSlotUpgradeable.getBooleanSlot(bytes32) (OVRLandContainer.sol#310-314) uses assembly
        - INLINE ASM (OVRLandContainer.sol#311-313)
StorageSlotUpgradeable.getBytes32Slot(bytes32) (OVRLandContainer.sol#319-323) uses assembly
        - INLINE ASM (OVRLandContainer.sol#320-322)
StorageSlotUpgradeable.getUint256Slot(bytes32) (OVRLandContainer.sol#328-332) uses assembly
        - INLINE ASM (OVRLandContainer.sol#329-331)
AddressUpgradeable.verifyCallResult(bool,bytes,string) (OVRLandContainer.sol#499-519) uses assembly
        - INLINE ASM (OVRLandContainer.sol#511-514)
ERC721Upgradeable._checkOnERC721Received(address,address,uint256,bytes) (OVRLandContainer.sol#1323-1344) uses assembly
        - INLINE ASM (OVRLandContainer.sol#1336-1338)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#assembly-usage
INFO:Detectors:
OVRLandContainer.landsFree(uint256[]) (OVRLandContainer.sol#2046-2081) compares to a boolean constant:
        -require(bool,string)(marketplace.landIsOnSelling(_landId[i]) == false,OVRLandContainer: One or more lands are on sellin
 (OVRLandContainer.sol#2052-2055)
OVRLandContainer.landsFree(uint256[]) (OVRLandContainer.sol#2046-2081) compares to a boolean constant:
        -require(bool,string)(renting.landIsOnRenting(_landId[i_scope_0]) == false,OVRLandContainer: One or more lands are on re
ing) (OVRLandContainer.sol#2061-2064)
OVRLandContainer.landsFree(uint256[]) (OVRLandContainer.sol#2046-2081) compares to a boolean constant:
        -require(bool,string)(marketplace.landIsOnSelling(_landId[i_scope_1]) == false,OVRLandContainer: One or more lands are o
selling) (OVRLandContainer.sol#2074-2077)
OVRLandContainer.landsFree(uint256[]) (OVRLandContainer.sol#2046-2081) compares to a boolean constant:
        -require(bool,string)(renting.landIsOnRenting(_landId[i_scope_1]) == false,OVRLandContainer: One or more lands are on re
ing) (OVRLandContainer.sol#2070-2073)
OVRLandContainer.landFree(uint256) (OVRLandContainer.sol#2087-2115) compares to a boolean constant:
        -require(bool,string)(marketplace.landIsOnSelling(_landId) == false,OVRLandContainer: One or more lands are on selling)
VRLandContainer.sol#2091-2094)
OVRLandContainer.landFree(uint256) (OVRLandContainer.sol#2087-2115) compares to a boolean constant:
        -require(bool,string)(marketplace.landIsOnSelling(_landId) == false,OVRLandContainer: One or more lands are on selling)
VRLandContainer.sol#2109-2112)
OVRLandContainer.landFree(uint256) (OVRLandContainer.sol#2087-2115) compares to a boolean constant:
        -require(bool,string)(renting.landIsOnRenting(_landId) == false,OVRLandContainer: One or more lands are on renting) (OVR
ndContainer.sol#2098-2101)
OVRLandContainer.landFree(uint256) (OVRLandContainer.sol#2087-2115) compares to a boolean constant:
```

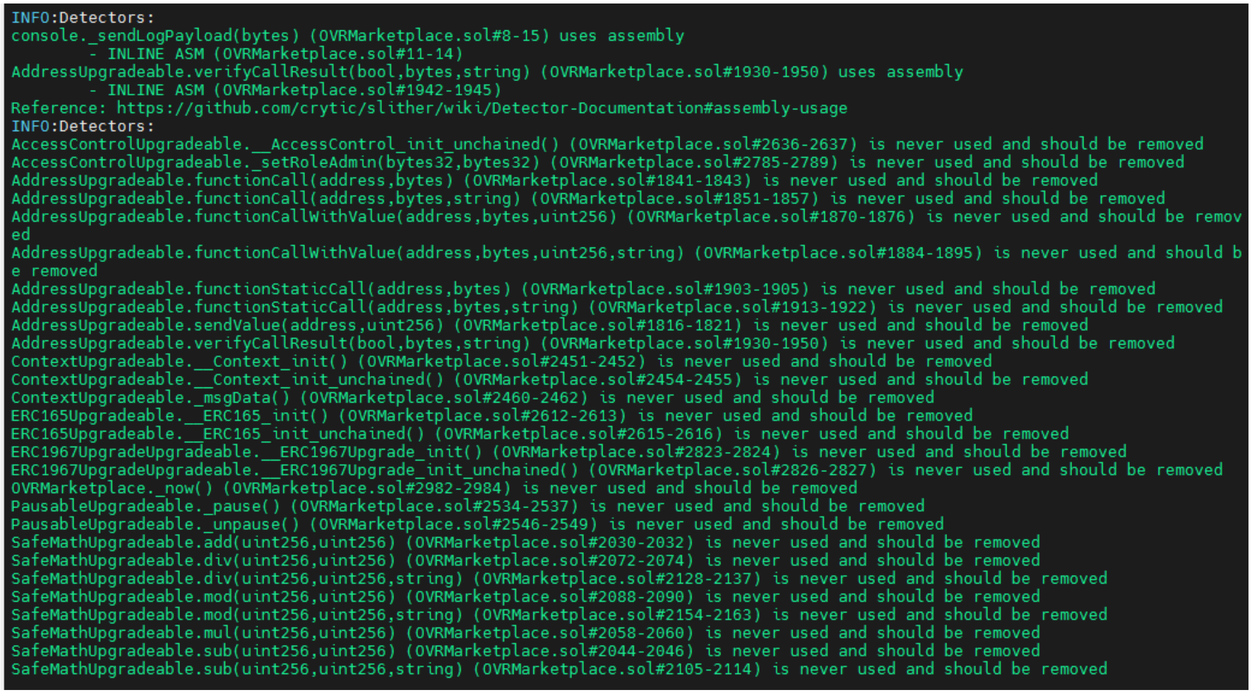


```
INFO:Detectors:
ERC1967UpgradeUpgradeable._ROLLBACK_SLOT (OVRLandContainer.sol#1856) is never used in OVRLandContainer (OVRLandContainer.sol#196
2447)
AccessControlUpgradeable.__gap (OVRLandContainer.sol#1683) is never used in OVRLandContainer (OVRLandContainer.sol#1965-2447)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#unused-state-variables
INFO:Detectors:
name() should be declared external:
        - ERC721Upgradeable.name() (OVRLandContainer.sol#1014-1016)
symbol() should be declared external:
        - ERC721Upgradeable.symbol() (OVRLandContainer.sol#1021-1023)
approve(address,uint256) should be declared external:
        - ERC721Upgradeable.approve(address,uint256) (OVRLandContainer.sol#1047-1057)
setApprovalForAll(address,bool) should be declared external:
        - ERC721Upgradeable.setApprovalForAll(address,bool) (OVRLandContainer.sol#1071-1073)
transferFrom(address,address,uint256) should be declared external:
        - ERC721Upgradeable.transferFrom(address,address,uint256) (OVRLandContainer.sol#1085-1094)
safeTransferFrom(address,address,uint256) should be declared external:
        - ERC721Upgradeable.safeTransferFrom(address,address,uint256) (OVRLandContainer.sol#1099-1105)
burn(uint256) should be declared external:
        - ERC721BurnableUpgradeable.burn(uint256) (OVRLandContainer.sol#1473-1477)
renounceRole(bytes32,address) should be declared external:
        - AccessControlUpgradeable.renounceRole(bytes32,address) (OVRLandContainer.sol#1615-1619)
tokenOfOwnerByIndex(address,uint256) should be declared external:
        - ERC721EnumerableUpgradeable.tokenOfOwnerByIndex(address,uint256) (OVRLandContainer.sol#1714-1717)
tokenByIndex(uint256) should be declared external:
        - ERC721EnumerableUpgradeable.tokenByIndex(uint256) (OVRLandContainer.sol#1729-1732)
ownerOfChild(uint256) should be declared external:
        - OVRLandContainer.ownerOfChild(uint256) (OVRLandContainer.sol#2155-2163)
childsOfParent(uint256) should be declared external:
        - OVRLandContainer.childsOfParent(uint256) (OVRLandContainer.sol#2168-2183)
setMarketplaceAddress(IMarketplace) should be declared external:
        - OVRLandContainer.setMarketplaceAddress(IMarketplace) (OVRLandContainer.sol#2188-2197)
setRentingAddress(IRenting) should be declared external:
        - OVRLandContainer.setRentingAddress(IRenting) (OVRLandContainer.sol#2202-2208)
removeLandFromContainer(uint256,uint256) should be declared external:
        - OVRLandContainer.removeLandFromContainer(uint256,uint256) (OVRLandContainer.sol#2213-2261)
addLandToContainer(uint256,uint256) should be declared external:
```

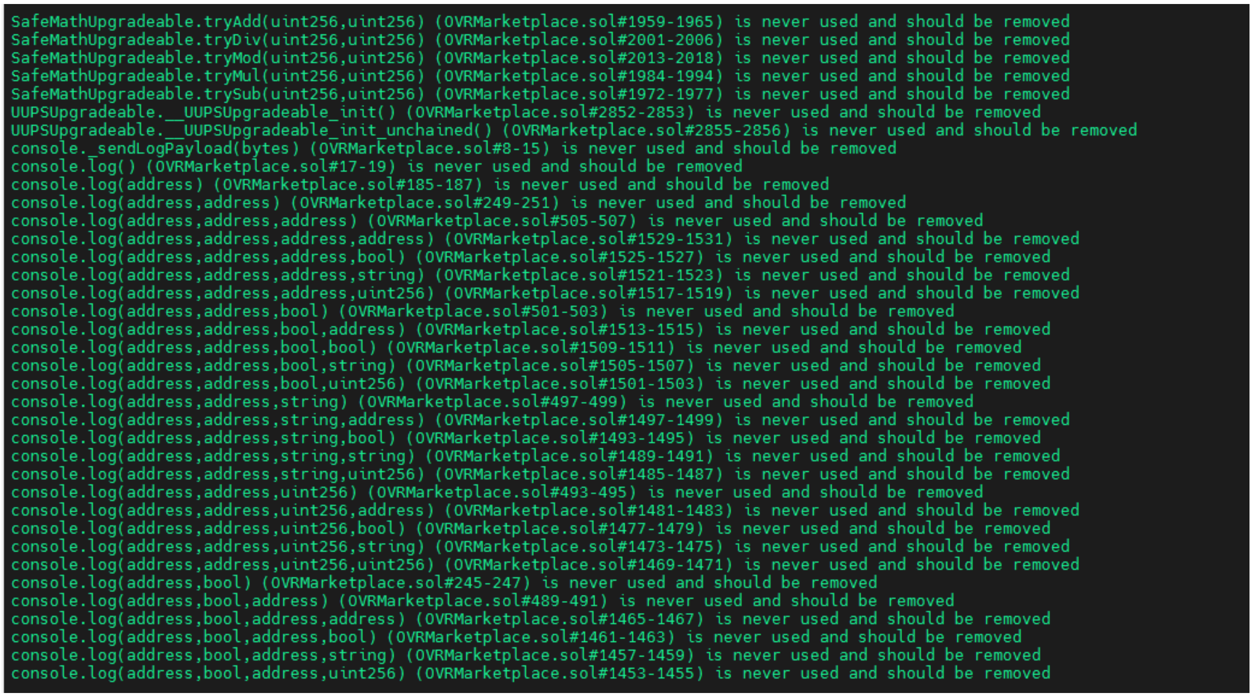


```
addLandToContainer(uint256,uint256) should be declared external:
        - OVRLandContainer.addLandToContainer(uint256,uint256) (OVRLandContainer.sol#2266-2294)
createContainer(uint256[]) should be declared external:
        - OVRLandContainer.createContainer(uint256[]) (OVRLandContainer.sol#2299-2318)
addURIEditor(address) should be declared external:
        - OVRLandContainer.addURIEditor(address) (OVRLandContainer.sol#2351-2353)
removeURIEditor(address) should be declared external:
        - OVRLandContainer.removeURIEditor(address) (OVRLandContainer.sol#2355-2360)
addUpgrader(address) should be declared external:
        - OVRLandContainer.addUpgrader(address) (OVRLandContainer.sol#2362-2367)
removeUpgrader(address) should be declared external:
        - OVRLandContainer.removeUpgrader(address) (OVRLandContainer.sol#2369-2374)
addAdminRole(address) should be declared external:
        - OVRLandContainer.addAdminRole(address) (OVRLandContainer.sol#2376-2380)
removeAdminRole(address) should be declared external:
        - OVRLandContainer.removeAdminRole(address) (OVRLandContainer.sol#2382-2389)
setOVRLandContainerURI(uint256,string) should be declared external:
        - OVRLandContainer.setOVRLandContainerURI(uint256,string) (OVRLandContainer.sol#2402-2407)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#public-function-that-could-be-declared-external
INFO:Slither:OVRLandContainer.sol analyzed (26 contracts with 75 detectors), 168 result(s) found
INFO:Slither:Use https://crytic.io/ to get access to additional detectors and Github integration
```

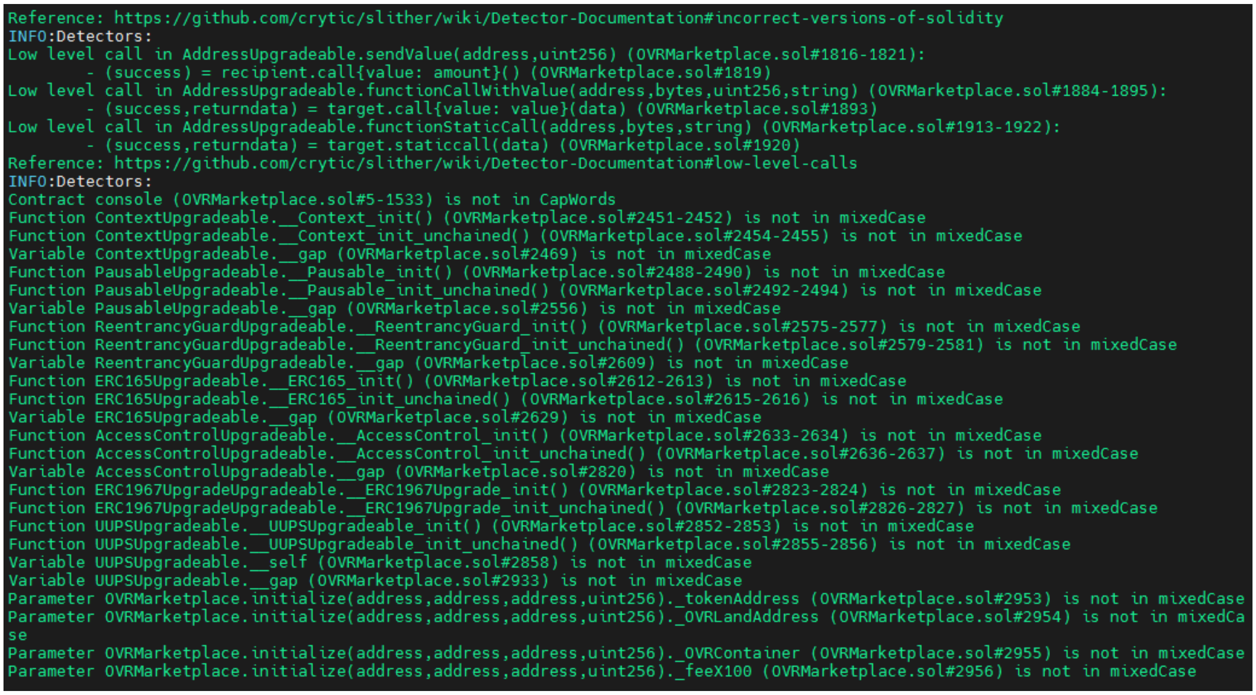
Slither log >> OVRMarketplace.sol



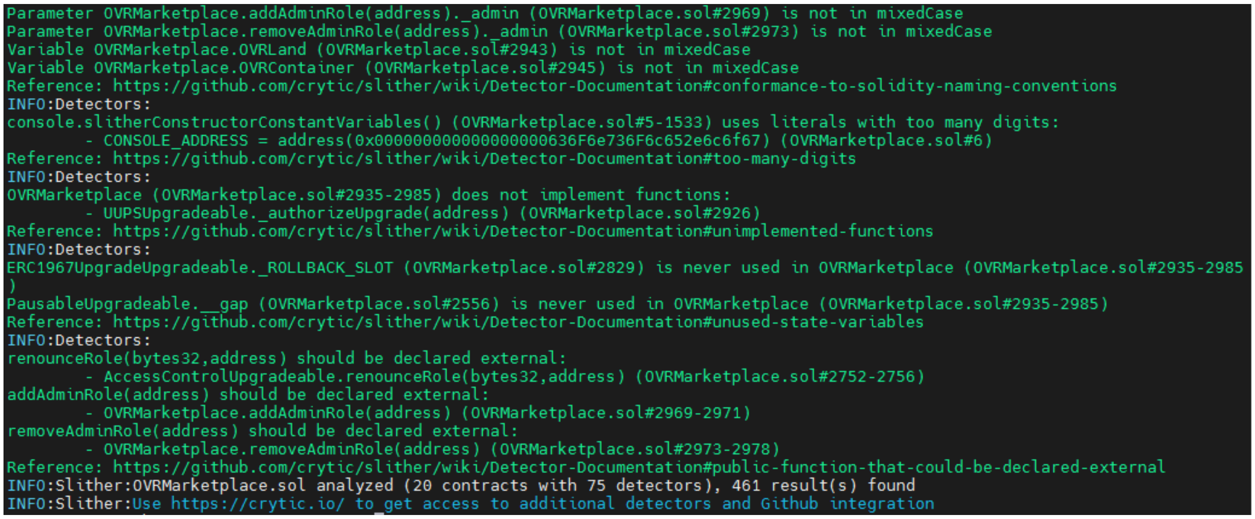
```
INFO:Detectors:
console._sendLogPayload(bytes) (OVRMarketplace.sol#8-15) uses assembly
        - INLINE ASM (OVRMarketplace.sol#11-14)
AddressUpgradeable.verifyCallResult(bool,bytes,string) (OVRMarketplace.sol#1930-1950) uses assembly
        - INLINE ASM (OVRMarketplace.sol#1942-1945)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#assembly-usage
INFO:Detectors:
AccessControlUpgradeable.__AccessControl_init_unchained() (OVRMarketplace.sol#2636-2637) is never used and should be removed
AccessControlUpgradeable._setRoleAdmin(bytes32,bytes32) (OVRMarketplace.sol#2785-2789) is never used and should be removed
AddressUpgradeable.functionCall(address,bytes) (OVRMarketplace.sol#1841-1843) is never used and should be removed
AddressUpgradeable.functionCall(address,bytes,string) (OVRMarketplace.sol#1851-1857) is never used and should be removed
AddressUpgradeable.functionCallWithValue(address,bytes,uint256) (OVRMarketplace.sol#1870-1876) is never used and should be remov
ed
AddressUpgradeable.functionCallWithValue(address,bytes,uint256,string) (OVRMarketplace.sol#1884-1895) is never used and should b
e removed
AddressUpgradeable.functionStaticCall(address,bytes) (OVRMarketplace.sol#1903-1905) is never used and should be removed
AddressUpgradeable.functionStaticCall(address,bytes,string) (OVRMarketplace.sol#1913-1922) is never used and should be removed
AddressUpgradeable.sendValue(address,uint256) (OVRMarketplace.sol#1816-1821) is never used and should be removed
AddressUpgradeable.verifyCallResult(bool,bytes,string) (OVRMarketplace.sol#1930-1950) is never used and should be removed
ContextUpgradeable.__Context_init() (OVRMarketplace.sol#2451-2452) is never used and should be removed
ContextUpgradeable.__Context_init_unchained() (OVRMarketplace.sol#2454-2455) is never used and should be removed
ContextUpgradeable._msgData() (OVRMarketplace.sol#2460-2462) is never used and should be removed
ERC165Upgradeable.__ERC165_init() (OVRMarketplace.sol#2612-2613) is never used and should be removed
ERC165Upgradeable.__ERC165_init_unchained() (OVRMarketplace.sol#2615-2616) is never used and should be removed
ERC1967UpgradeUpgradeable.__ERC1967Upgrade_init() (OVRMarketplace.sol#2823-2824) is never used and should be removed
ERC1967UpgradeUpgradeable.__ERC1967Upgrade_init_unchained() (OVRMarketplace.sol#2826-2827) is never used and should be removed
OVRMarketplace._now() (OVRMarketplace.sol#2982-2984) is never used and should be removed
PausableUpgradeable._pause() (OVRMarketplace.sol#2534-2537) is never used and should be removed
PausableUpgradeable._unpause() (OVRMarketplace.sol#2546-2549) is never used and should be removed
SafeMathUpgradeable.add(uint256,uint256) (OVRMarketplace.sol#2030-2032) is never used and should be removed
SafeMathUpgradeable.div(uint256,uint256) (OVRMarketplace.sol#2072-2074) is never used and should be removed
SafeMathUpgradeable.div(uint256,uint256,string) (OVRMarketplace.sol#2128-2137) is never used and should be removed
SafeMathUpgradeable.mod(uint256,uint256) (OVRMarketplace.sol#2088-2090) is never used and should be removed
SafeMathUpgradeable.mod(uint256,uint256,string) (OVRMarketplace.sol#2154-2163) is never used and should be removed
SafeMathUpgradeable.mul(uint256,uint256) (OVRMarketplace.sol#2058-2060) is never used and should be removed
SafeMathUpgradeable.sub(uint256,uint256) (OVRMarketplace.sol#2044-2046) is never used and should be removed
SafeMathUpgradeable.sub(uint256,uint256,string) (OVRMarketplace.sol#2105-2114) is never used and should be removed
```



```
SafeMathUpgradeable.tryAdd(uint256,uint256) (OVRMarketplace.sol#1959-1965) is never used and should be removed
SafeMathUpgradeable.tryDiv(uint256,uint256) (OVRMarketplace.sol#2001-2006) is never used and should be removed
SafeMathUpgradeable.tryMod(uint256,uint256) (OVRMarketplace.sol#2013-2018) is never used and should be removed
SafeMathUpgradeable.tryMul(uint256,uint256) (OVRMarketplace.sol#1984-1994) is never used and should be removed
SafeMathUpgradeable.trySub(uint256,uint256) (OVRMarketplace.sol#1972-1977) is never used and should be removed
UUPSUpgradeable.__UUPSUpgradeable_init() (OVRMarketplace.sol#2852-2853) is never used and should be removed
UUPSUpgradeable.__UUPSUpgradeable_init_unchained() (OVRMarketplace.sol#2855-2856) is never used and should be removed
console._sendLogPayload(bytes) (OVRMarketplace.sol#8-15) is never used and should be removed
console.log() (OVRMarketplace.sol#17-19) is never used and should be removed
console.log(address) (OVRMarketplace.sol#185-187) is never used and should be removed
console.log(address,address) (OVRMarketplace.sol#249-251) is never used and should be removed
console.log(address,address,address) (OVRMarketplace.sol#505-507) is never used and should be removed
console.log(address,address,address,address) (OVRMarketplace.sol#1529-1531) is never used and should be removed
console.log(address,address,address,bool) (OVRMarketplace.sol#1525-1527) is never used and should be removed
console.log(address,address,address,string) (OVRMarketplace.sol#1521-1523) is never used and should be removed
console.log(address,address,address,uint256) (OVRMarketplace.sol#1517-1519) is never used and should be removed
console.log(address,address,bool) (OVRMarketplace.sol#501-503) is never used and should be removed
console.log(address,address,bool,address) (OVRMarketplace.sol#1513-1515) is never used and should be removed
console.log(address,address,bool,bool) (OVRMarketplace.sol#1509-1511) is never used and should be removed
console.log(address,address,bool,string) (OVRMarketplace.sol#1505-1507) is never used and should be removed
console.log(address,address,bool,uint256) (OVRMarketplace.sol#1501-1503) is never used and should be removed
console.log(address,address,string) (OVRMarketplace.sol#497-499) is never used and should be removed
console.log(address,address,string,address) (OVRMarketplace.sol#1497-1499) is never used and should be removed
console.log(address,address,string,bool) (OVRMarketplace.sol#1493-1495) is never used and should be removed
console.log(address,address,string,string) (OVRMarketplace.sol#1489-1491) is never used and should be removed
console.log(address,address,string,uint256) (OVRMarketplace.sol#1485-1487) is never used and should be removed
console.log(address,address,uint256) (OVRMarketplace.sol#493-495) is never used and should be removed
console.log(address,address,uint256,address) (OVRMarketplace.sol#1481-1483) is never used and should be removed
console.log(address,address,uint256,bool) (OVRMarketplace.sol#1477-1479) is never used and should be removed
console.log(address,address,uint256,string) (OVRMarketplace.sol#1473-1475) is never used and should be removed
console.log(address,address,uint256,uint256) (OVRMarketplace.sol#1469-1471) is never used and should be removed
console.log(address,bool) (OVRMarketplace.sol#245-247) is never used and should be removed
console.log(address,bool,address) (OVRMarketplace.sol#489-491) is never used and should be removed
console.log(address,bool,address,address) (OVRMarketplace.sol#1465-1467) is never used and should be removed
console.log(address,bool,address,bool) (OVRMarketplace.sol#1461-1463) is never used and should be removed
console.log(address,bool,address,string) (OVRMarketplace.sol#1457-1459) is never used and should be removed
console.log(address,bool,address,uint256) (OVRMarketplace.sol#1453-1455) is never used and should be removed
```

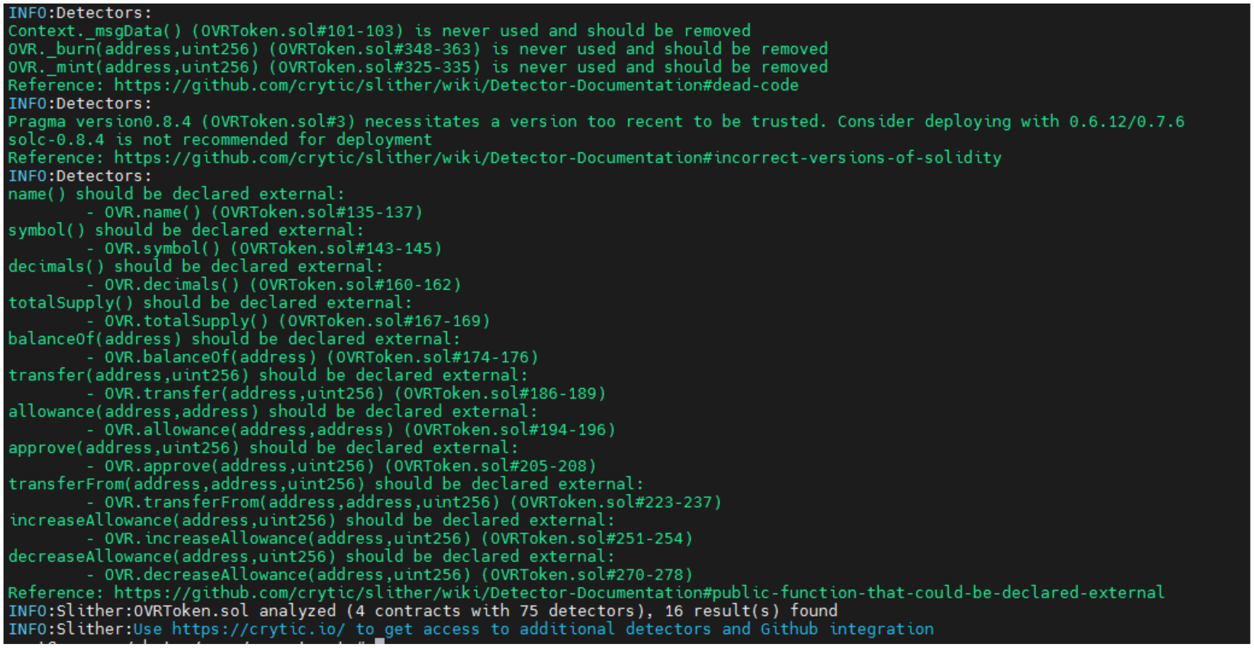


```
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-versions-of-solidity
INFO:Detectors:
Low level call in AddressUpgradeable.sendValue(address,uint256) (OVRMarketplace.sol#1816-1821):
        - (success) = recipient.call{value: amount}() (OVRMarketplace.sol#1819)
Low level call in AddressUpgradeable.functionCallWithValue(address,bytes,uint256,string) (OVRMarketplace.sol#1884-1895):
        - (success,returndata) = target.call{value: value}(data) (OVRMarketplace.sol#1893)
Low level call in AddressUpgradeable.functionStaticCall(address,bytes,string) (OVRMarketplace.sol#1913-1922):
        - (success,returndata) = target.staticcall(data) (OVRMarketplace.sol#1920)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#low-level-calls
INFO:Detectors:
Contract console (OVRMarketplace.sol#5-1533) is not in CapWords
Function ContextUpgradeable.__Context_init() (OVRMarketplace.sol#2451-2452) is not in mixedCase
Function ContextUpgradeable.__Context_init_unchained() (OVRMarketplace.sol#2454-2455) is not in mixedCase
Variable ContextUpgradeable.__gap (OVRMarketplace.sol#2469) is not in mixedCase
Function PausableUpgradeable.__Pausable_init() (OVRMarketplace.sol#2488-2490) is not in mixedCase
Function PausableUpgradeable.__Pausable_init_unchained() (OVRMarketplace.sol#2492-2494) is not in mixedCase
Variable PausableUpgradeable.__gap (OVRMarketplace.sol#2556) is not in mixedCase
Function ReentrancyGuardUpgradeable.__ReentrancyGuard_init() (OVRMarketplace.sol#2575-2577) is not in mixedCase
Function ReentrancyGuardUpgradeable.__ReentrancyGuard_init_unchained() (OVRMarketplace.sol#2579-2581) is not in mixedCase
Variable ReentrancyGuardUpgradeable.__gap (OVRMarketplace.sol#2609) is not in mixedCase
Function ERC165Upgradeable.__ERC165_init() (OVRMarketplace.sol#2612-2613) is not in mixedCase
Function ERC165Upgradeable.__ERC165_init_unchained() (OVRMarketplace.sol#2615-2616) is not in mixedCase
Variable ERC165Upgradeable.__gap (OVRMarketplace.sol#2629) is not in mixedCase
Function AccessControlUpgradeable.__AccessControl_init() (OVRMarketplace.sol#2633-2634) is not in mixedCase
Function AccessControlUpgradeable.__AccessControl_init_unchained() (OVRMarketplace.sol#2636-2637) is not in mixedCase
Variable AccessControlUpgradeable.__gap (OVRMarketplace.sol#2820) is not in mixedCase
Function ERC1967UpgradeUpgradeable.__ERC1967Upgrade_init() (OVRMarketplace.sol#2823-2824) is not in mixedCase
Function ERC1967UpgradeUpgradeable.__ERC1967Upgrade_init_unchained() (OVRMarketplace.sol#2826-2827) is not in mixedCase
Function UUPSUpgradeable.__UUPSUpgradeable_init() (OVRMarketplace.sol#2852-2853) is not in mixedCase
Function UUPSUpgradeable.__UUPSUpgradeable_init_unchained() (OVRMarketplace.sol#2855-2856) is not in mixedCase
Variable UUPSUpgradeable.__self (OVRMarketplace.sol#2858) is not in mixedCase
Variable UUPSUpgradeable.__gap (OVRMarketplace.sol#2933) is not in mixedCase
Parameter OVRMarketplace.initialize(address,address,address,uint256)._tokenAddress (OVRMarketplace.sol#2953) is not in mixedCase
Parameter OVRMarketplace.initialize(address,address,address,uint256)._OVRLandAddress (OVRMarketplace.sol#2954) is not in mixedCa
se
Parameter OVRMarketplace.initialize(address,address,address,uint256)._OVRContainer (OVRMarketplace.sol#2955) is not in mixedCase
Parameter OVRMarketplace.initialize(address,address,address,uint256)._feeX100 (OVRMarketplace.sol#2956) is not in mixedCase
```



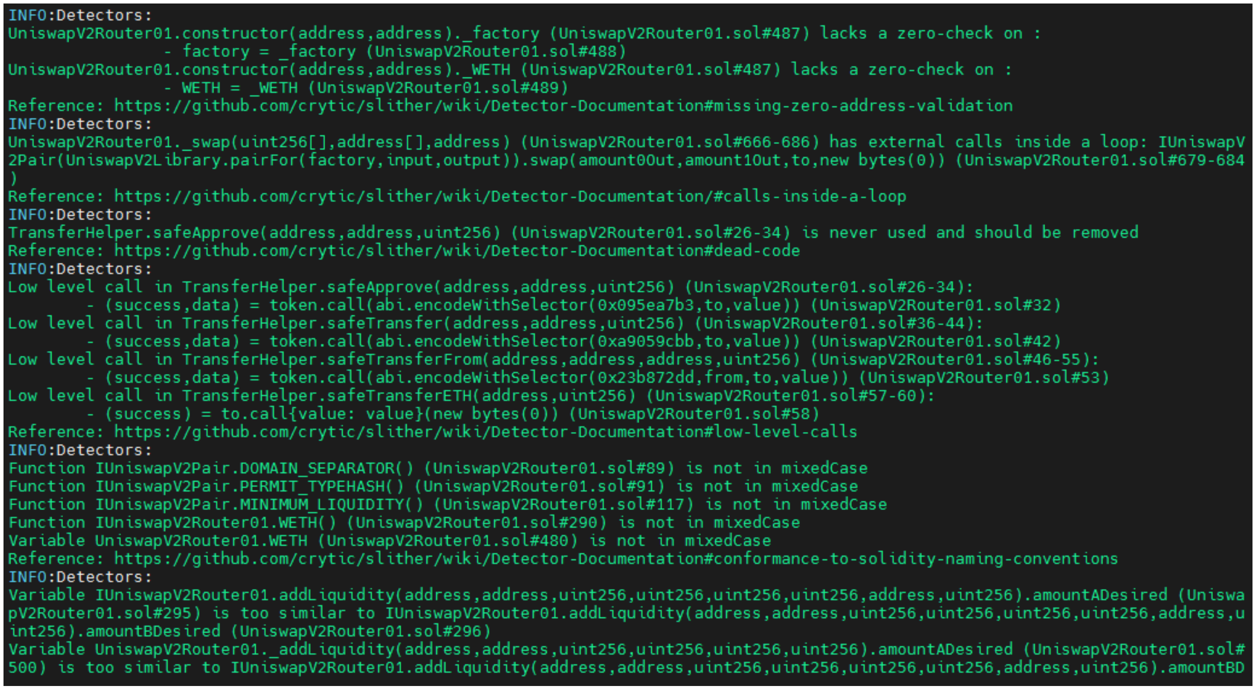
```
Parameter OVRMarketplace.addAdminRole(address)._admin (OVRMarketplace.sol#2969) is not in mixedCase
Parameter OVRMarketplace.removeAdminRole(address)._admin (OVRMarketplace.sol#2973) is not in mixedCase
Variable OVRMarketplace.OVRLand (OVRMarketplace.sol#2943) is not in mixedCase
Variable OVRMarketplace.OVRContainer (OVRMarketplace.sol#2945) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-solidity-naming-conventions
INFO:Detectors:
console.slitherConstructorConstantVariables() (OVRMarketplace.sol#5-1533) uses literals with too many digits:
        - CONSOLE_ADDRESS = address(0x000000000000000000636F6e736F6c652e6c6f67) (OVRMarketplace.sol#6)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#too-many-digits
INFO:Detectors:
OVRMarketplace (OVRMarketplace.sol#2935-2985) does not implement functions:
        - UUPSUpgradeable._authorizeUpgrade(address) (OVRMarketplace.sol#2926)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#unimplemented-functions
INFO:Detectors:
ERC1967UpgradeUpgradeable._ROLLBACK_SLOT (OVRMarketplace.sol#2829) is never used in OVRMarketplace (OVRMarketplace.sol#2935-2985
)
PausableUpgradeable.__gap (OVRMarketplace.sol#2556) is never used in OVRMarketplace (OVRMarketplace.sol#2935-2985)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#unused-state-variables
INFO:Detectors:
renounceRole(bytes32,address) should be declared external:
        - AccessControlUpgradeable.renounceRole(bytes32,address) (OVRMarketplace.sol#2752-2756)
addAdminRole(address) should be declared external:
        - OVRMarketplace.addAdminRole(address) (OVRMarketplace.sol#2969-2971)
removeAdminRole(address) should be declared external:
        - OVRMarketplace.removeAdminRole(address) (OVRMarketplace.sol#2973-2978)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#public-function-that-could-be-declared-external
INFO:Slither:OVRMarketplace.sol analyzed (20 contracts with 75 detectors), 461 result(s) found
INFO:Slither:Use https://crytic.io/ to get access to additional detectors and Github integration
```

Slither log >> OVRToken.sol

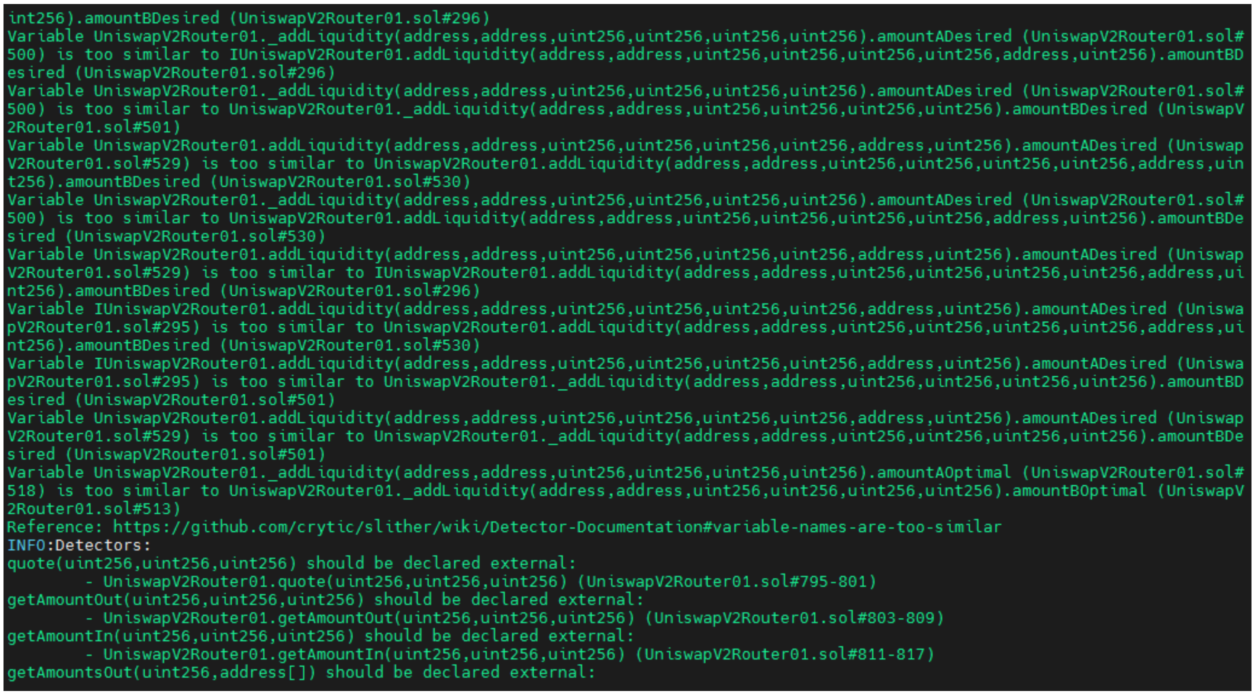


```
INFO:Detectors:
Context._msgData() (OVRToken.sol#101-103) is never used and should be removed
OVR._burn(address,uint256) (OVRToken.sol#348-363) is never used and should be removed
OVR._mint(address,uint256) (OVRToken.sol#325-335) is never used and should be removed
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dead-code
INFO:Detectors:
Pragma version0.8.4 (OVRToken.sol#3) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
solc-0.8.4 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-versions-of-solidity
INFO:Detectors:
name() should be declared external:
        - OVR.name() (OVRToken.sol#135-137)
symbol() should be declared external:
        - OVR.symbol() (OVRToken.sol#143-145)
decimals() should be declared external:
        - OVR.decimals() (OVRToken.sol#160-162)
totalSupply() should be declared external:
        - OVR.totalSupply() (OVRToken.sol#167-169)
balanceOf(address) should be declared external:
        - OVR.balanceOf(address) (OVRToken.sol#174-176)
transfer(address,uint256) should be declared external:
        - OVR.transfer(address,uint256) (OVRToken.sol#186-189)
allowance(address,address) should be declared external:
        - OVR.allowance(address,address) (OVRToken.sol#194-196)
approve(address,uint256) should be declared external:
        - OVR.approve(address,uint256) (OVRToken.sol#205-208)
transferFrom(address,address,uint256) should be declared external:
        - OVR.transferFrom(address,address,uint256) (OVRToken.sol#223-237)
increaseAllowance(address,uint256) should be declared external:
        - OVR.increaseAllowance(address,uint256) (OVRToken.sol#251-254)
decreaseAllowance(address,uint256) should be declared external:
        - OVR.decreaseAllowance(address,uint256) (OVRToken.sol#270-278)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#public-function-that-could-be-declared-external
INFO:Slither:OVRToken.sol analyzed (4 contracts with 75 detectors), 16 result(s) found
INFO:Slither:Use https://crytic.io/ to get access to additional detectors and Github integration
```

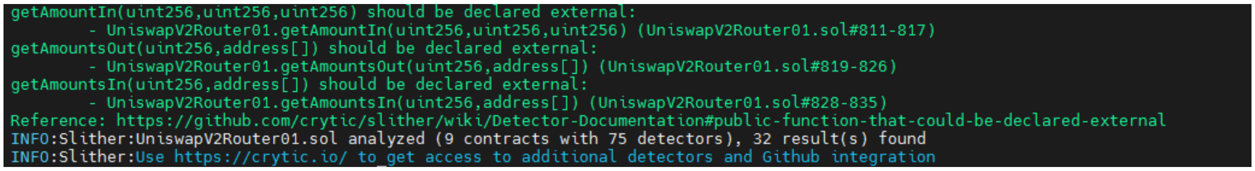
Slither log >> Uniswapv2router.sol



```
INFO:Detectors:
UniswapV2Router01.constructor(address,address)._factory (UniswapV2Router01.sol#487) lacks a zero-check on :
                - factory = _factory (UniswapV2Router01.sol#488)
UniswapV2Router01.constructor(address,address)._WETH (UniswapV2Router01.sol#487) lacks a zero-check on :
                - WETH = _WETH (UniswapV2Router01.sol#489)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#missing-zero-address-validation
INFO:Detectors:
UniswapV2Router01._swap(uint256[],address[],address) (UniswapV2Router01.sol#666-686) has external calls inside a loop: IUniswapV
2Pair(UniswapV2Library.pairFor(factory,input,output)).swap(amount0Out,amount1Out,to,new bytes(0)) (UniswapV2Router01.sol#679-684
)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation/#calls-inside-a-loop
INFO:Detectors:
TransferHelper.safeApprove(address,address,uint256) (UniswapV2Router01.sol#26-34) is never used and should be removed
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dead-code
INFO:Detectors:
Low level call in TransferHelper.safeApprove(address,address,uint256) (UniswapV2Router01.sol#26-34):
        - (success,data) = token.call(abi.encodeWithSelector(0x095ea7b3,to,value)) (UniswapV2Router01.sol#32)
Low level call in TransferHelper.safeTransfer(address,address,uint256) (UniswapV2Router01.sol#36-44):
        - (success,data) = token.call(abi.encodeWithSelector(0xa9059cbb,to,value)) (UniswapV2Router01.sol#42)
Low level call in TransferHelper.safeTransferFrom(address,address,address,uint256) (UniswapV2Router01.sol#46-55):
        - (success,data) = token.call(abi.encodeWithSelector(0x23b872dd,from,to,value)) (UniswapV2Router01.sol#53)
Low level call in TransferHelper.safeTransferETH(address,uint256) (UniswapV2Router01.sol#57-60):
        - (success) = to.call{value: value}(new bytes(0)) (UniswapV2Router01.sol#58)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#low-level-calls
INFO:Detectors:
Function IUniswapV2Pair.DOMAIN_SEPARATOR() (UniswapV2Router01.sol#89) is not in mixedCase
Function IUniswapV2Pair.PERMIT_TYPEHASH() (UniswapV2Router01.sol#91) is not in mixedCase
Function IUniswapV2Pair.MINIMUM_LIQUIDITY() (UniswapV2Router01.sol#117) is not in mixedCase
Function IUniswapV2Router01.WETH() (UniswapV2Router01.sol#290) is not in mixedCase
Variable UniswapV2Router01.WETH (UniswapV2Router01.sol#480) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-solidity-naming-conventions
INFO:Detectors:
Variable IUniswapV2Router01.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uint256).amountADesired (Uniswa
pV2Router01.sol#295) is too similar to IUniswapV2Router01.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,u
int256).amountBDesired (UniswapV2Router01.sol#296)
Variable UniswapV2Router01._addLiquidity(address,address,uint256,uint256,uint256,uint256).amountADesired (UniswapV2Router01.sol#
500) is too similar to IUniswapV2Router01.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uint256).amountBD
```



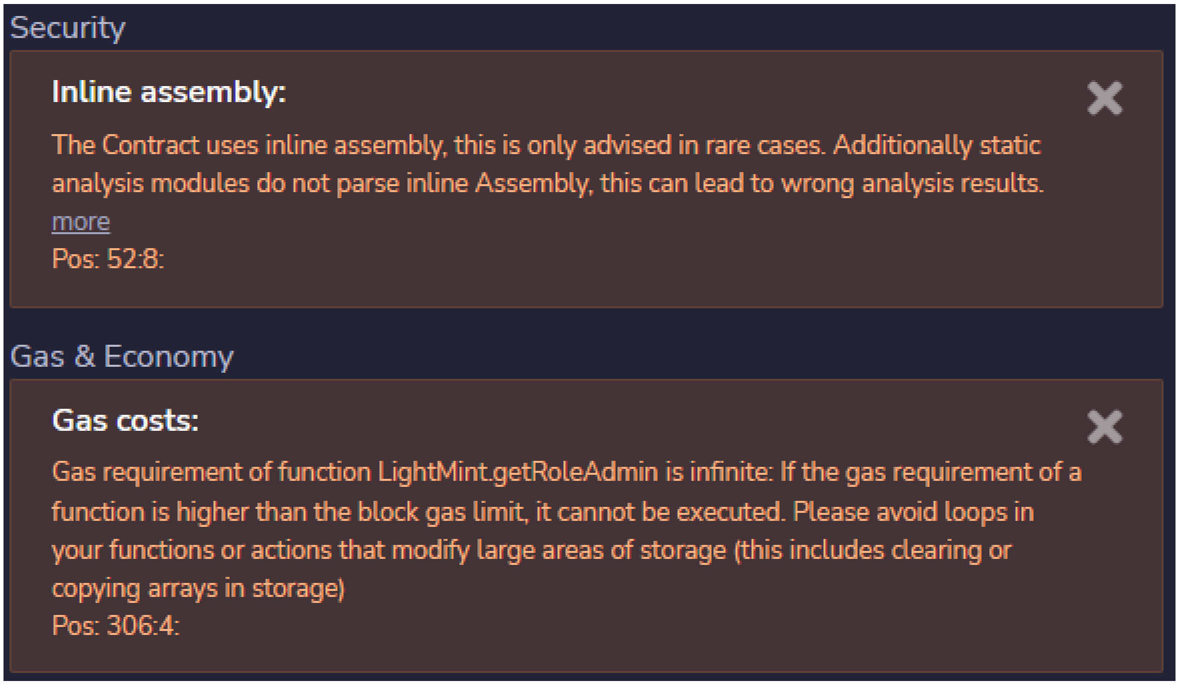
```
int256).amountBDesired (UniswapV2Router01.sol#296)
Variable UniswapV2Router01._addLiquidity(address,address,uint256,uint256,uint256,uint256).amountADesired (UniswapV2Router01.sol#
500) is too similar to IUniswapV2Router01.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uint256).amountBD
esired (UniswapV2Router01.sol#296)
Variable UniswapV2Router01._addLiquidity(address,address,uint256,uint256,uint256,uint256).amountADesired (UniswapV2Router01.sol#
500) is too similar to UniswapV2Router01._addLiquidity(address,address,uint256,uint256,uint256,uint256).amountBDesired (UniswapV
2Router01.sol#501)
Variable UniswapV2Router01.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uint256).amountADesired (Uniswap
V2Router01.sol#529) is too similar to UniswapV2Router01.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uin
t256).amountBDesired (UniswapV2Router01.sol#530)
Variable UniswapV2Router01._addLiquidity(address,address,uint256,uint256,uint256,uint256).amountADesired (UniswapV2Router01.sol#
500) is too similar to UniswapV2Router01.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uint256).amountBDe
sired (UniswapV2Router01.sol#530)
Variable UniswapV2Router01.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uint256).amountADesired (Uniswap
V2Router01.sol#529) is too similar to IUniswapV2Router01.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,ui
nt256).amountBDesired (UniswapV2Router01.sol#296)
Variable IUniswapV2Router01.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uint256).amountADesired (Uniswa
pV2Router01.sol#295) is too similar to UniswapV2Router01.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,ui
nt256).amountBDesired (UniswapV2Router01.sol#530)
Variable IUniswapV2Router01.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uint256).amountADesired (Uniswa
pV2Router01.sol#295) is too similar to UniswapV2Router01._addLiquidity(address,address,uint256,uint256,uint256,uint256).amountBD
esired (UniswapV2Router01.sol#501)
Variable UniswapV2Router01.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uint256).amountADesired (Uniswap
V2Router01.sol#529) is too similar to UniswapV2Router01._addLiquidity(address,address,uint256,uint256,uint256,uint256).amountBDe
sired (UniswapV2Router01.sol#501)
Variable UniswapV2Router01._addLiquidity(address,address,uint256,uint256,uint256,uint256).amountAOptimal (UniswapV2Router01.sol#
518) is too similar to UniswapV2Router01._addLiquidity(address,address,uint256,uint256,uint256,uint256).amountBOptimal (UniswapV
2Router01.sol#513)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#variable-names-are-too-similar
INFO:Detectors:
quote(uint256,uint256,uint256) should be declared external:
        - UniswapV2Router01.quote(uint256,uint256,uint256) (UniswapV2Router01.sol#795-801)
getAmountOut(uint256,uint256,uint256) should be declared external:
        - UniswapV2Router01.getAmountOut(uint256,uint256,uint256) (UniswapV2Router01.sol#803-809)
getAmountIn(uint256,uint256,uint256) should be declared external:
        - UniswapV2Router01.getAmountIn(uint256,uint256,uint256) (UniswapV2Router01.sol#811-817)
getAmountsOut(uint256,address[]) should be declared external:
```



```
getAmountIn(uint256,uint256,uint256) should be declared external:
        - UniswapV2Router01.getAmountIn(uint256,uint256,uint256) (UniswapV2Router01.sol#811-817)
getAmountsOut(uint256,address[]) should be declared external:
        - UniswapV2Router01.getAmountsOut(uint256,address[]) (UniswapV2Router01.sol#819-826)
getAmountsIn(uint256,address[]) should be declared external:
        - UniswapV2Router01.getAmountsIn(uint256,address[]) (UniswapV2Router01.sol#828-835)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#public-function-that-could-be-declared-external
INFO:Slither:UniswapV2Router01.sol analyzed (9 contracts with 75 detectors), 32 result(s) found
INFO:Slither:Use https://crytic.io/ to get access to additional detectors and Github integration
```

Solidity Static Analysis

LightMint.sol



Security

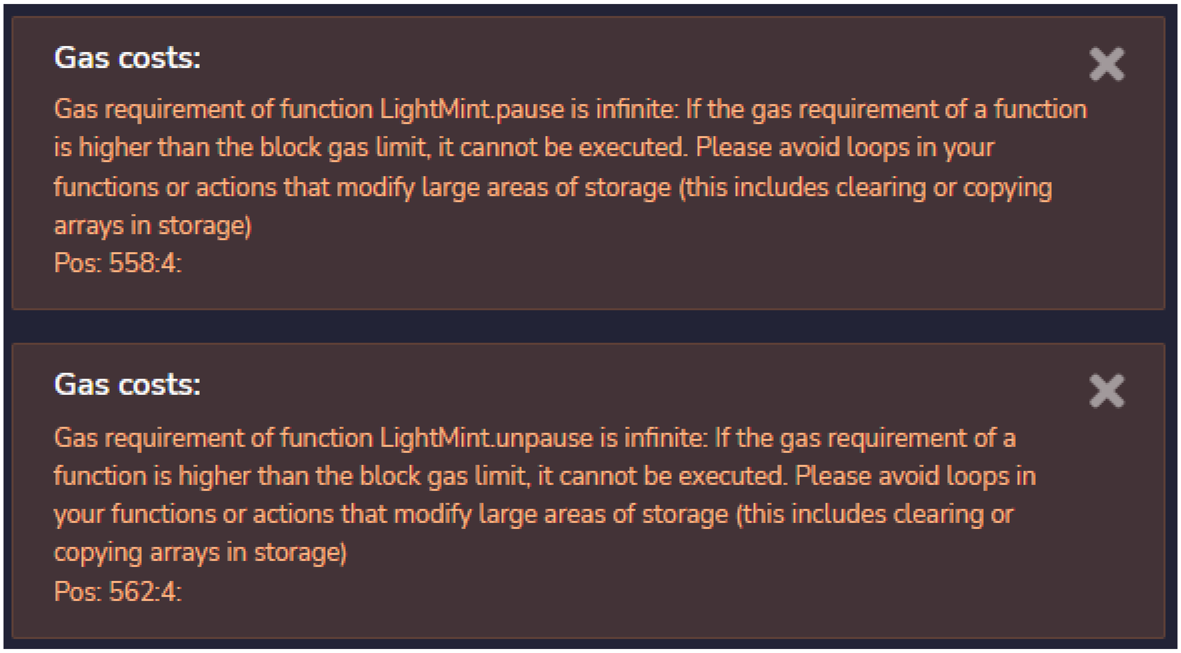
Inline assembly:

The Contract uses inline assembly, this is only advised in rare cases. Additionally static analysis modules do not parse inline Assembly, this can lead to wrong analysis results.
more
Pos: 52:8:

Gas & Economy

Gas costs:

Gas requirement of function LightMint.getRoleAdmin is infinite: If the gas requirement of a function is higher than the block gas limit, it cannot be executed. Please avoid loops in your functions or actions that modify large areas of storage (this includes clearing or copying arrays in storage)
Pos: 306:4:

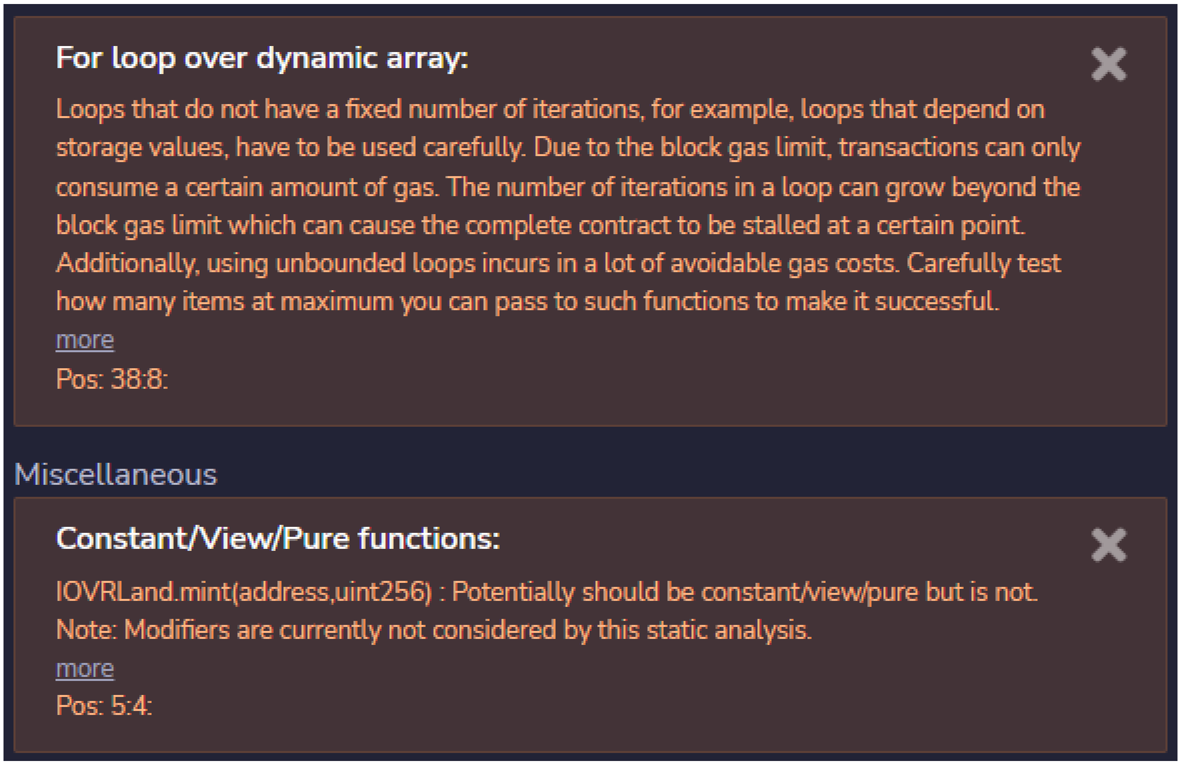


Gas costs:

Gas requirement of function LightMint.pause is infinite: If the gas requirement of a function is higher than the block gas limit, it cannot be executed. Please avoid loops in your functions or actions that modify large areas of storage (this includes clearing or copying arrays in storage)
Pos: 558:4:

Gas costs:

Gas requirement of function LightMint.unpause is infinite: If the gas requirement of a function is higher than the block gas limit, it cannot be executed. Please avoid loops in your functions or actions that modify large areas of storage (this includes clearing or copying arrays in storage)
Pos: 562:4:



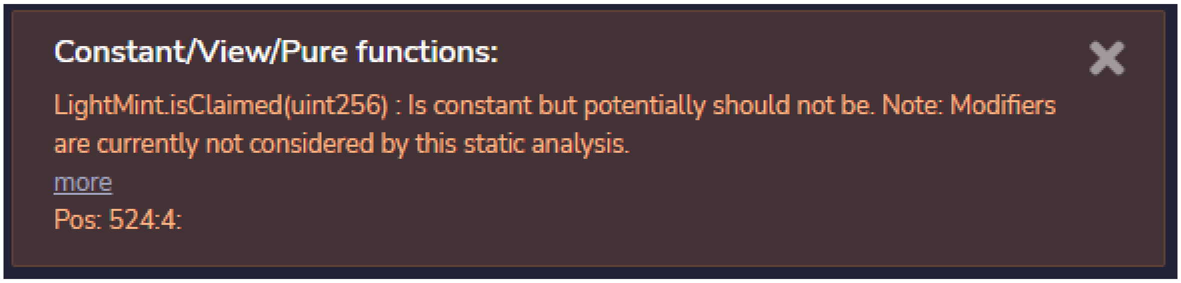
For loop over dynamic array:

Loops that do not have a fixed number of iterations, for example, loops that depend on storage values, have to be used carefully. Due to the block gas limit, transactions can only consume a certain amount of gas. The number of iterations in a loop can grow beyond the block gas limit which can cause the complete contract to be stalled at a certain point. Additionally, using unbounded loops incurs in a lot of avoidable gas costs. Carefully test how many items at maximum you can pass to such functions to make it successful.
more
Pos: 38:8:

Miscellaneous

Constant/View/Pure functions:

IOVRLand.mint(address,uint256) : Potentially should be constant/view/pure but is not. Note: Modifiers are currently not considered by this static analysis.
more
Pos: 5:4:

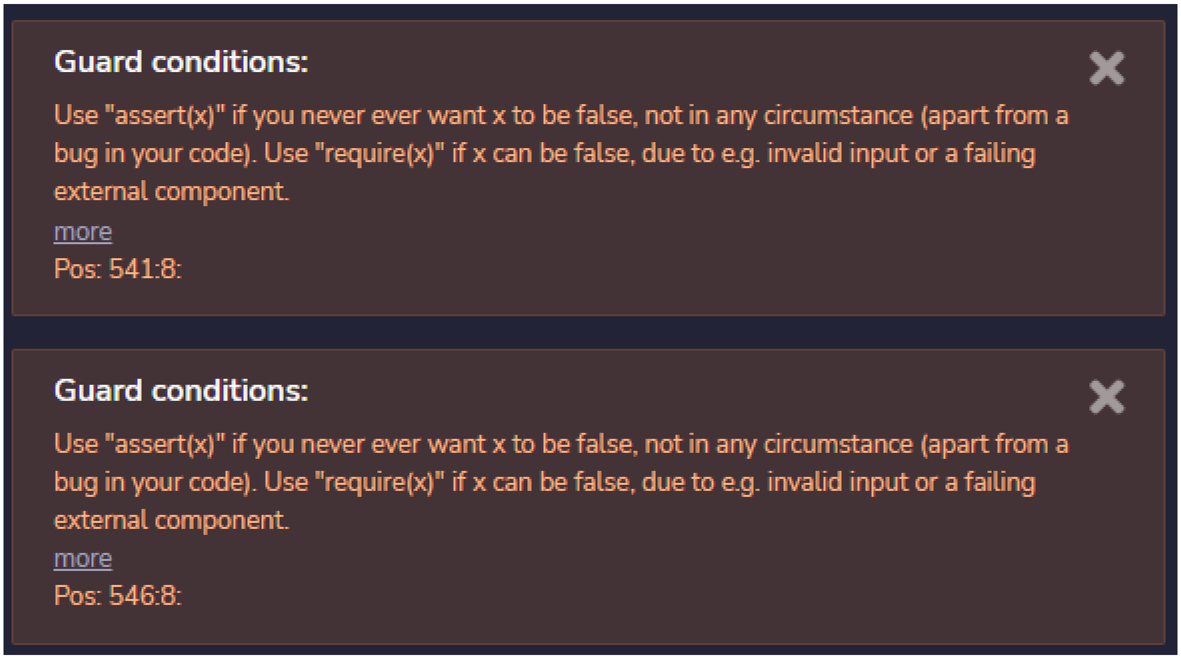


Constant/View/Pure functions:

LightMint.isClaimed(uint256) : Is constant but potentially should not be. Note: Modifiers are currently not considered by this static analysis.

more

Pos: 524:4:



Guard conditions:

Use "assert(x)" if you never ever want x to be false, not in any circumstance (apart from a bug in your code). Use "require(x)" if x can be false, due to e.g. invalid input or a failing external component.

more

Pos: 541:8:

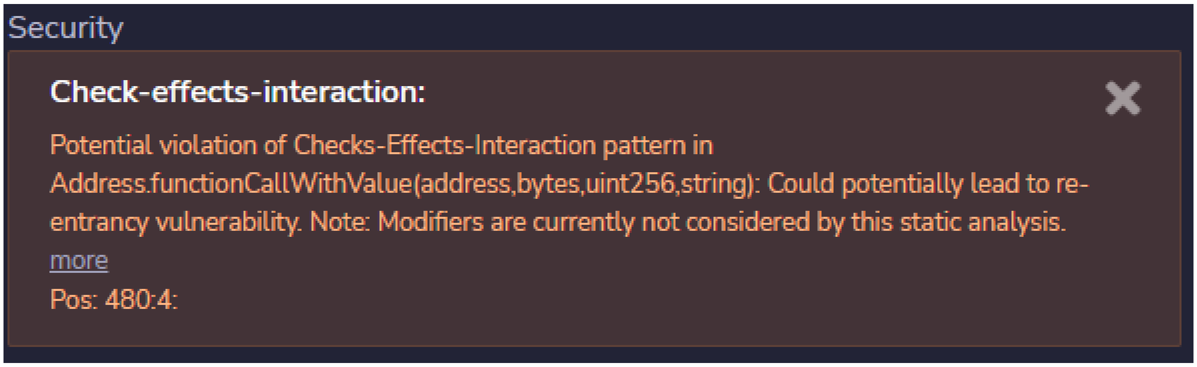
Guard conditions:

Use "assert(x)" if you never ever want x to be false, not in any circumstance (apart from a bug in your code). Use "require(x)" if x can be false, due to e.g. invalid input or a failing external component.

more

Pos: 546:8:

OVRLand.sol



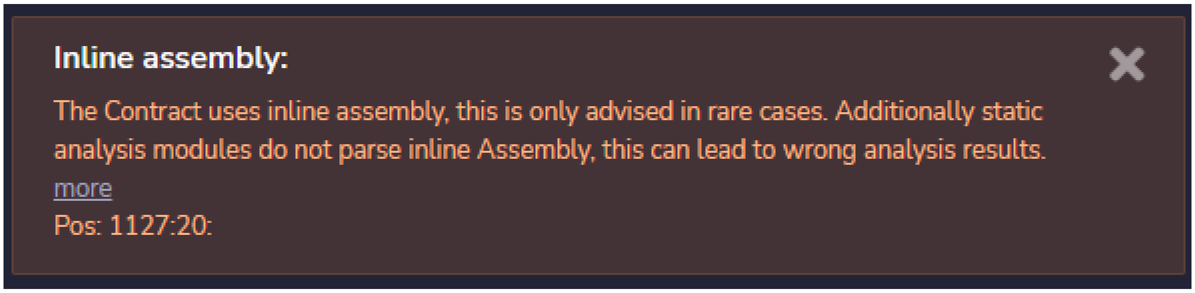
Security

Check-effects-interaction:

Potential violation of Checks-Effects-Interaction pattern in Address.functionCallWithValue(address,bytes,uint256,string): Could potentially lead to re-entrancy vulnerability. Note: Modifiers are currently not considered by this static analysis.

more

Pos: 480:4:

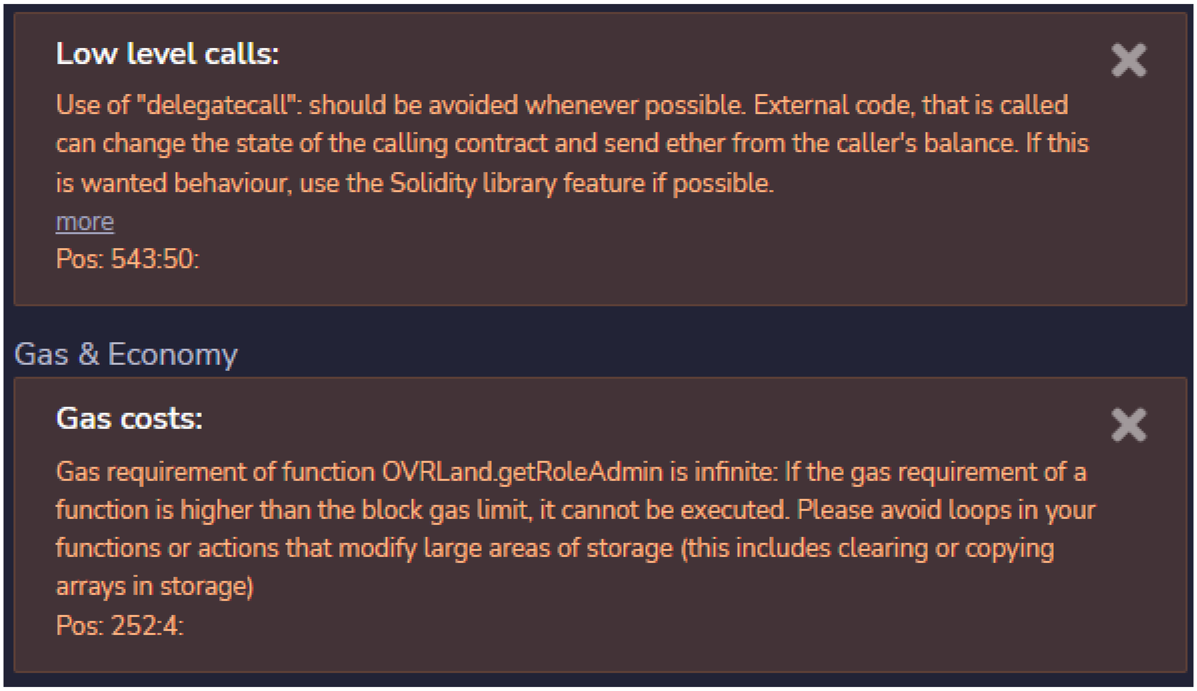


Inline assembly:

The Contract uses inline assembly, this is only advised in rare cases. Additionally static analysis modules do not parse inline Assembly, this can lead to wrong analysis results.

more

Pos: 1127:20:



Low level calls:

Use of "delegatecall": should be avoided whenever possible. External code, that is called can change the state of the calling contract and send ether from the caller's balance. If this is wanted behaviour, use the Solidity library feature if possible.

more

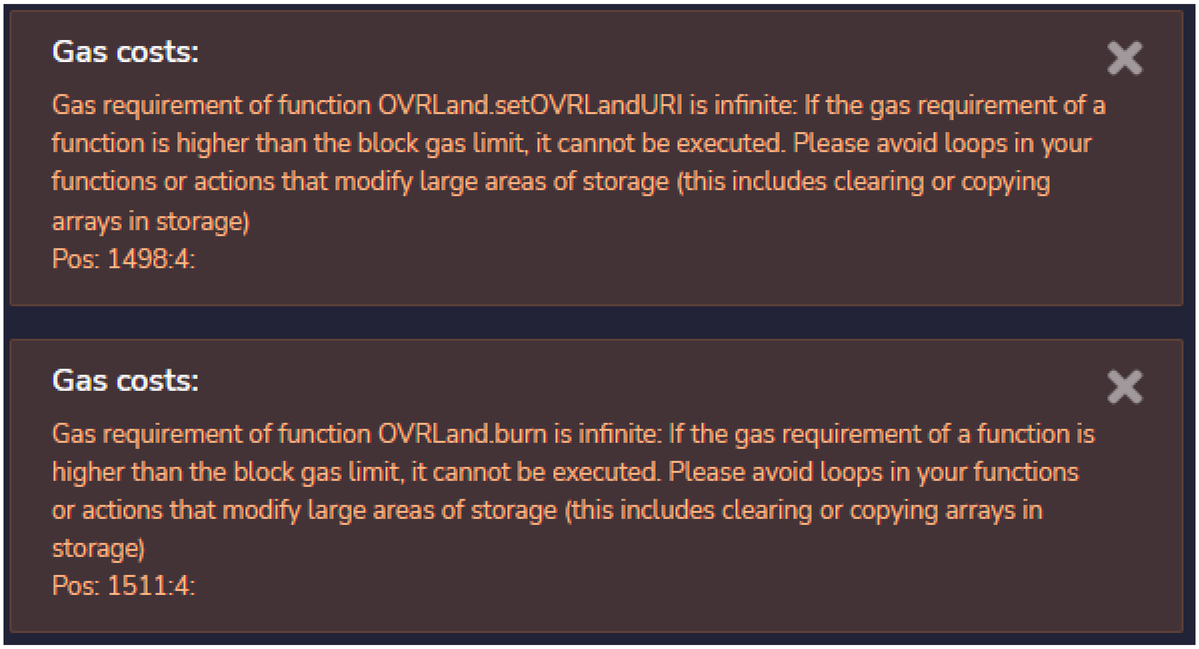
Pos: 543:50:

Gas & Economy

Gas costs:

Gas requirement of function OVRLand.getRoleAdmin is infinite: If the gas requirement of a function is higher than the block gas limit, it cannot be executed. Please avoid loops in your functions or actions that modify large areas of storage (this includes clearing or copying arrays in storage)

Pos: 252:4:

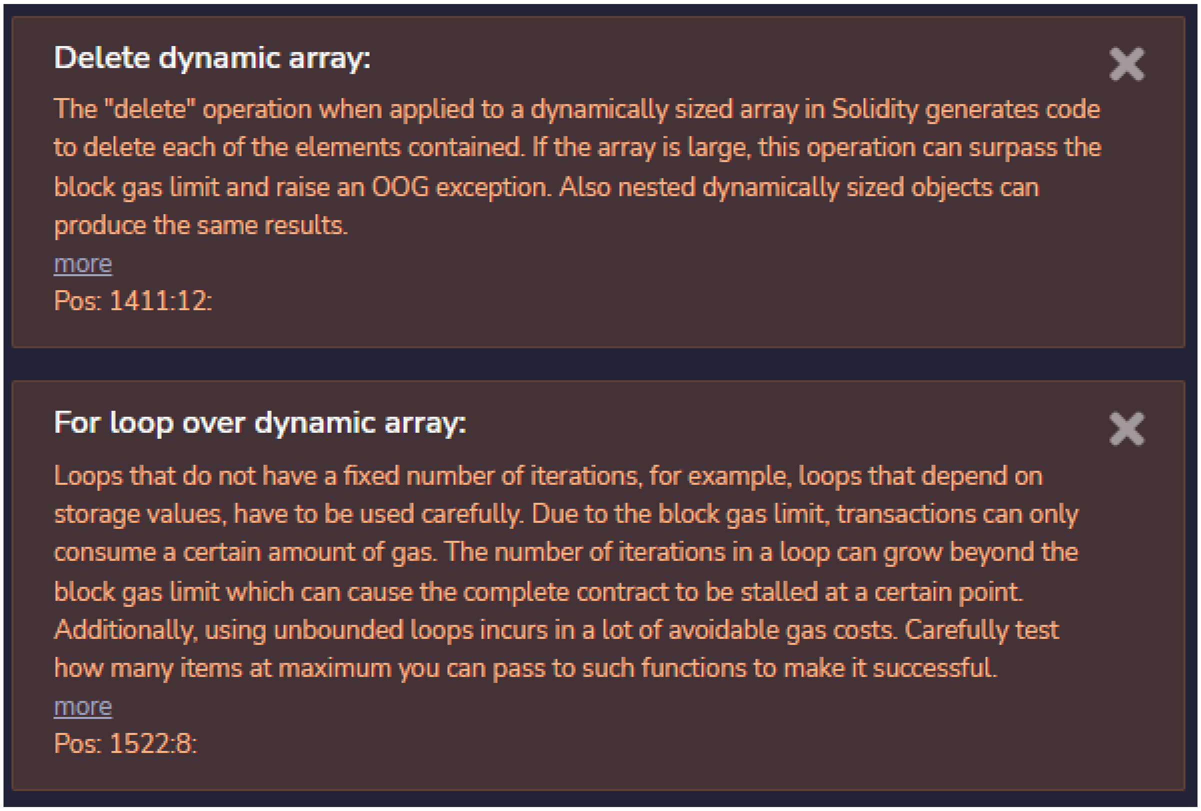


Gas costs:

Gas requirement of function OVRLand.setOVRLandURI is infinite: If the gas requirement of a function is higher than the block gas limit, it cannot be executed. Please avoid loops in your functions or actions that modify large areas of storage (this includes clearing or copying arrays in storage)
Pos: 1498:4:

Gas costs:

Gas requirement of function OVRLand.burn is infinite: If the gas requirement of a function is higher than the block gas limit, it cannot be executed. Please avoid loops in your functions or actions that modify large areas of storage (this includes clearing or copying arrays in storage)
Pos: 1511:4:

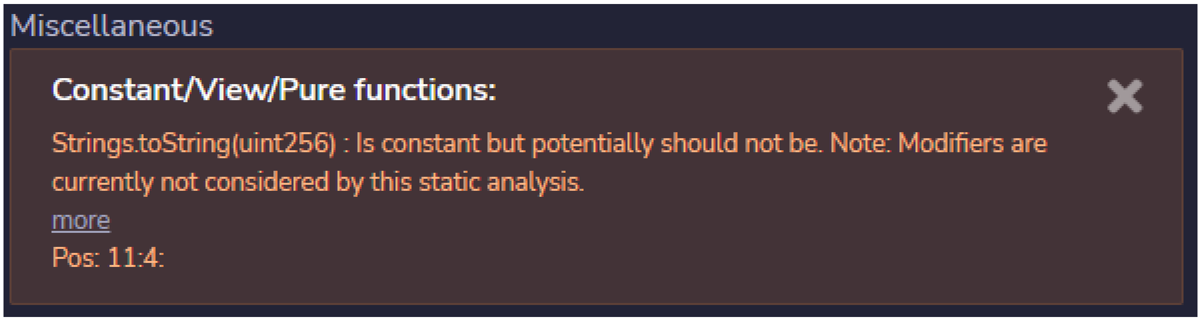


Delete dynamic array:

The "delete" operation when applied to a dynamically sized array in Solidity generates code to delete each of the elements contained. If the array is large, this operation can surpass the block gas limit and raise an OOG exception. Also nested dynamically sized objects can produce the same results.
more
Pos: 1411:12:

For loop over dynamic array:

Loops that do not have a fixed number of iterations, for example, loops that depend on storage values, have to be used carefully. Due to the block gas limit, transactions can only consume a certain amount of gas. The number of iterations in a loop can grow beyond the block gas limit which can cause the complete contract to be stalled at a certain point. Additionally, using unbounded loops incurs in a lot of avoidable gas costs. Carefully test how many items at maximum you can pass to such functions to make it successful.
more
Pos: 1522:8:



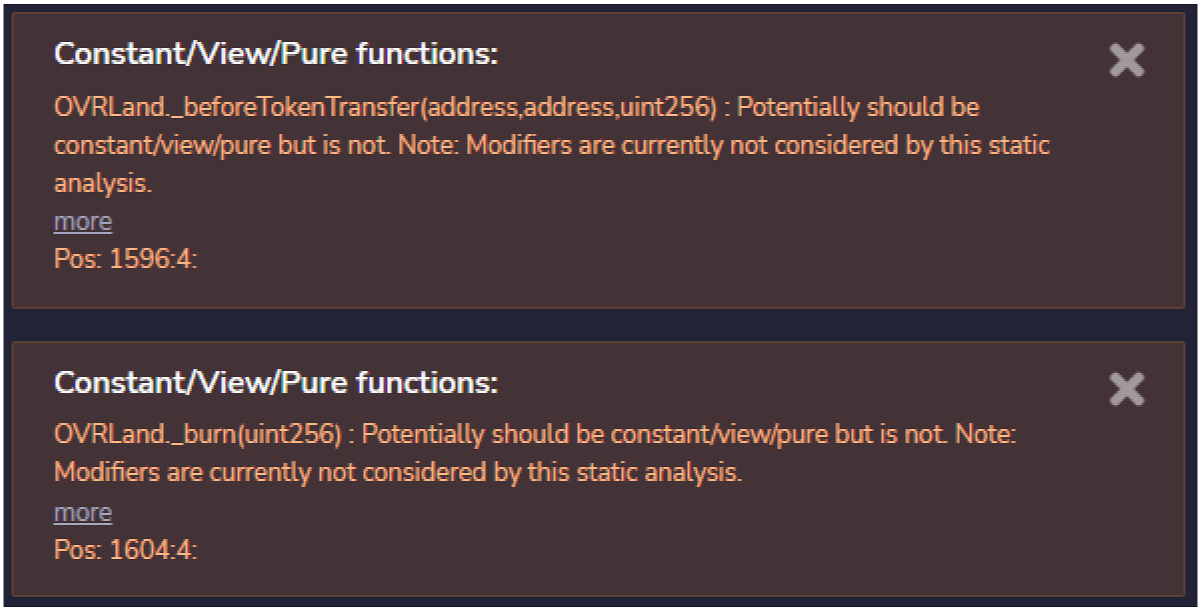
Miscellaneous

Constant/View/Pure functions:

Strings.toString(uint256) : Is constant but potentially should not be. Note: Modifiers are currently not considered by this static analysis.

more

Pos: 11:4:



Constant/View/Pure functions:

OVRLand._beforeTokenTransfer(address,address,uint256) : Potentially should be constant/view/pure but is not. Note: Modifiers are currently not considered by this static analysis.

more

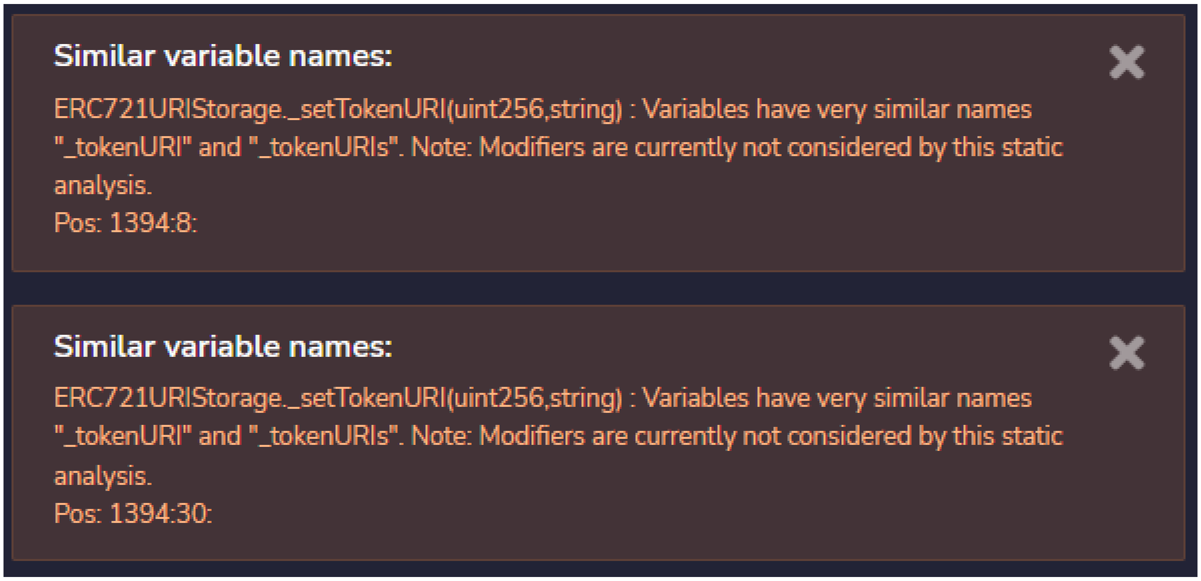
Pos: 1596:4:

Constant/View/Pure functions:

OVRLand._burn(uint256) : Potentially should be constant/view/pure but is not. Note: Modifiers are currently not considered by this static analysis.

more

Pos: 1604:4:

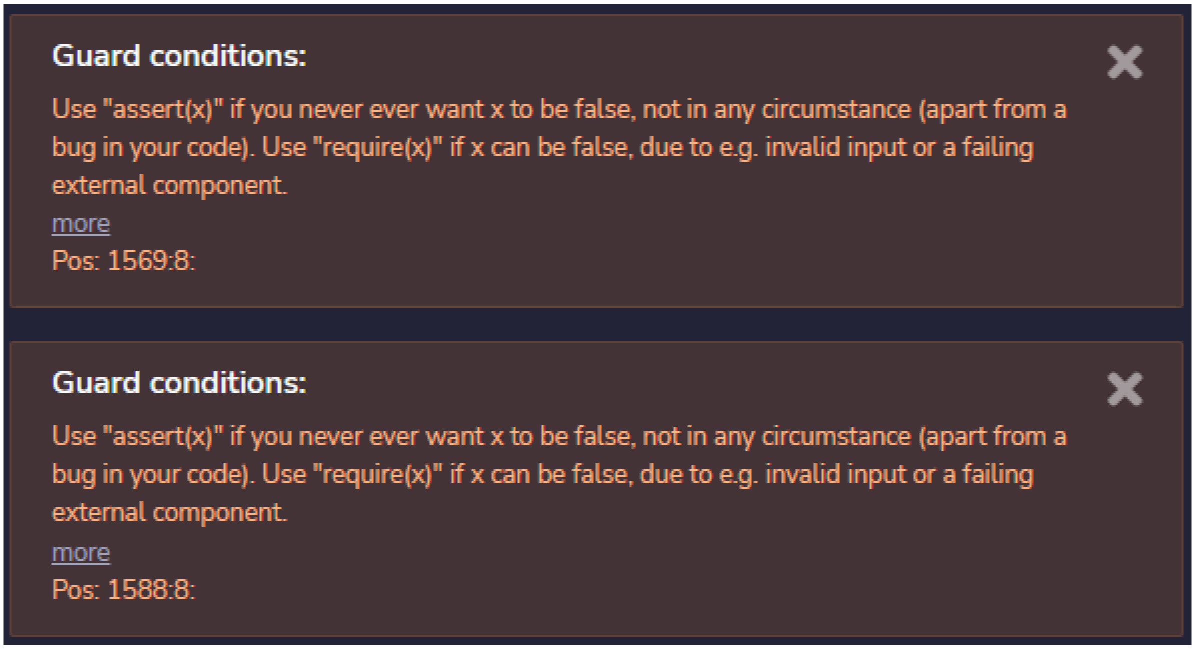


Similar variable names:

ERC721URIStorage._setTokenURI(uint256,string) : Variables have very similar names "_tokenURI" and "_tokenURIs". Note: Modifiers are currently not considered by this static analysis.
Pos: 1394:8:

Similar variable names:

ERC721URIStorage._setTokenURI(uint256,string) : Variables have very similar names "_tokenURI" and "_tokenURIs". Note: Modifiers are currently not considered by this static analysis.
Pos: 1394:30:

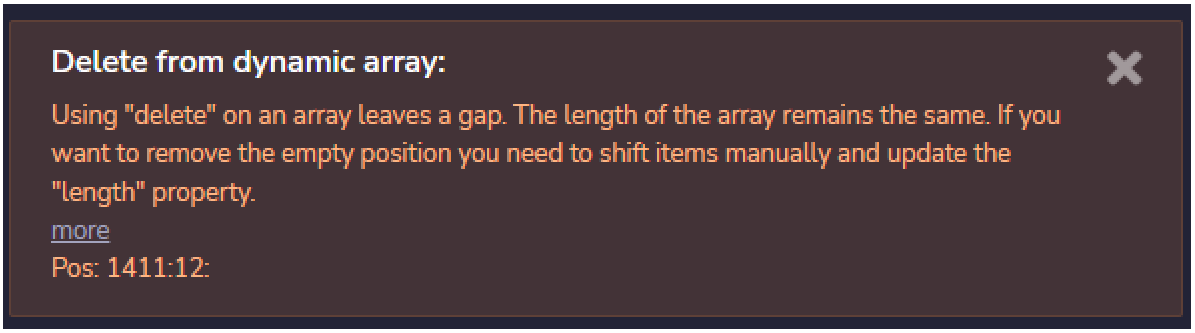


Guard conditions:

Use "assert(x)" if you never ever want x to be false, not in any circumstance (apart from a bug in your code). Use "require(x)" if x can be false, due to e.g. invalid input or a failing external component.
more
Pos: 1569:8:

Guard conditions:

Use "assert(x)" if you never ever want x to be false, not in any circumstance (apart from a bug in your code). Use "require(x)" if x can be false, due to e.g. invalid input or a failing external component.
more
Pos: 1588:8:



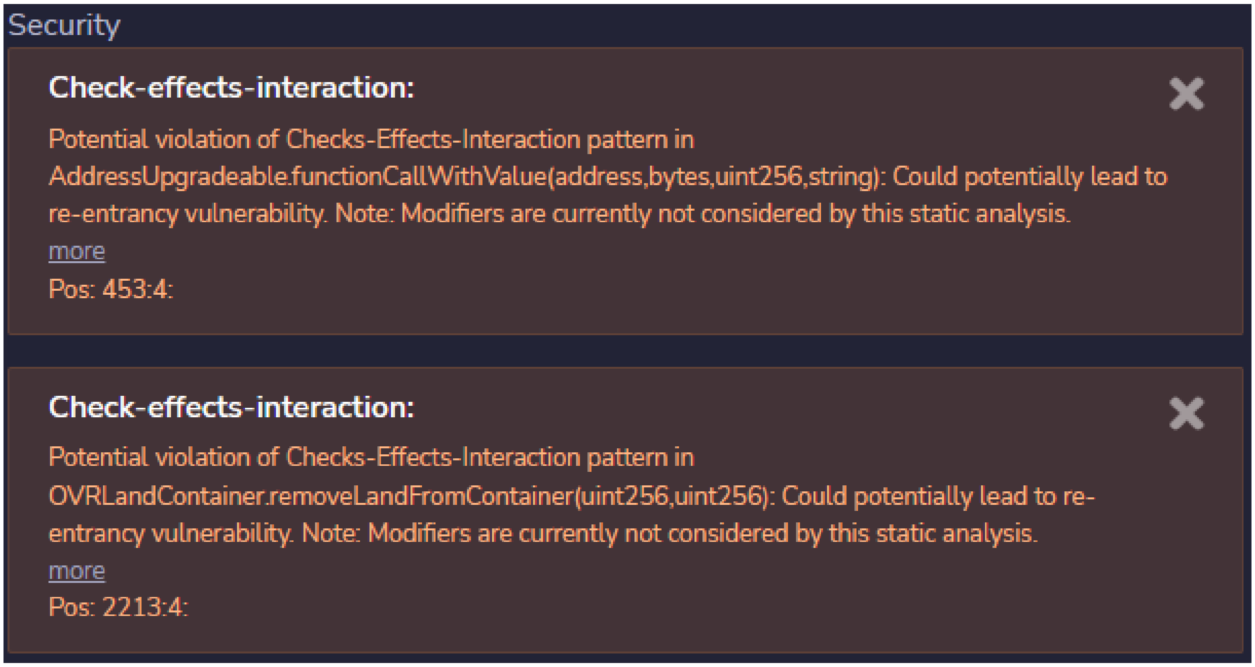
Delete from dynamic array:

Using "delete" on an array leaves a gap. The length of the array remains the same. If you want to remove the empty position you need to shift items manually and update the "length" property.

more

Pos: 1411:12:

OVRLandContainer.sol



Security

Check-effects-interaction:

Potential violation of Checks-Effects-Interaction pattern in AddressUpgradeable.functionCallWithValue(address,bytes,uint256,string): Could potentially lead to re-entrancy vulnerability. Note: Modifiers are currently not considered by this static analysis.

more

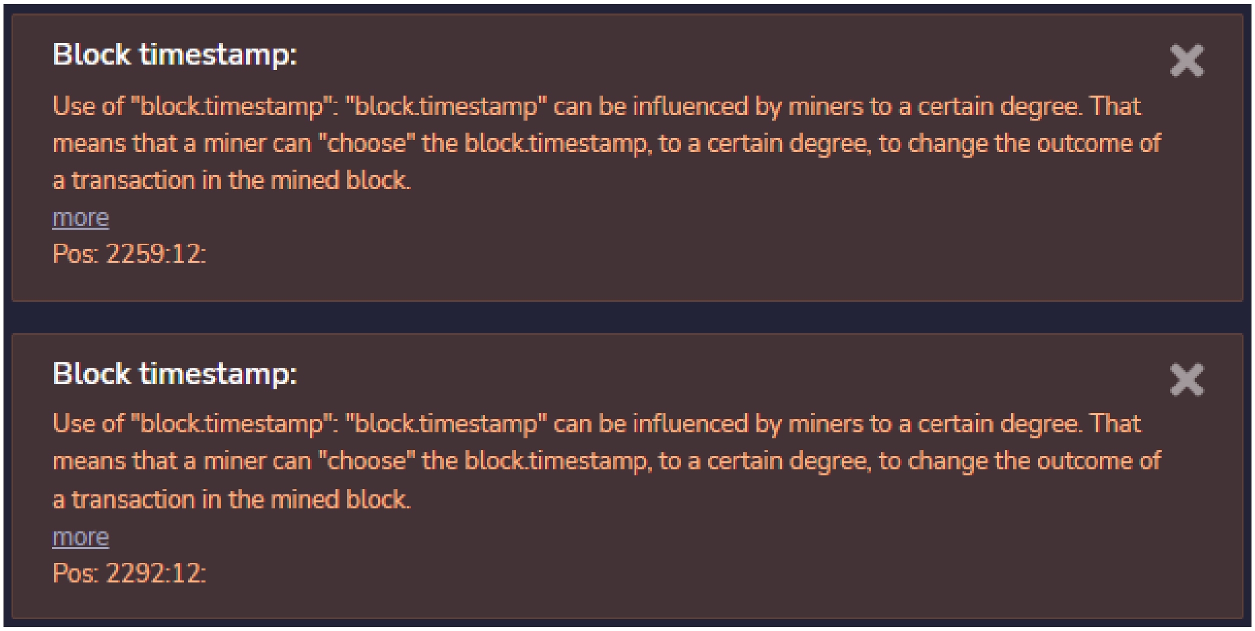
Pos: 453:4:

Check-effects-interaction:

Potential violation of Checks-Effects-Interaction pattern in OVRLandContainer.removeLandFromContainer(uint256,uint256): Could potentially lead to re-entrancy vulnerability. Note: Modifiers are currently not considered by this static analysis.

more

Pos: 2213:4:

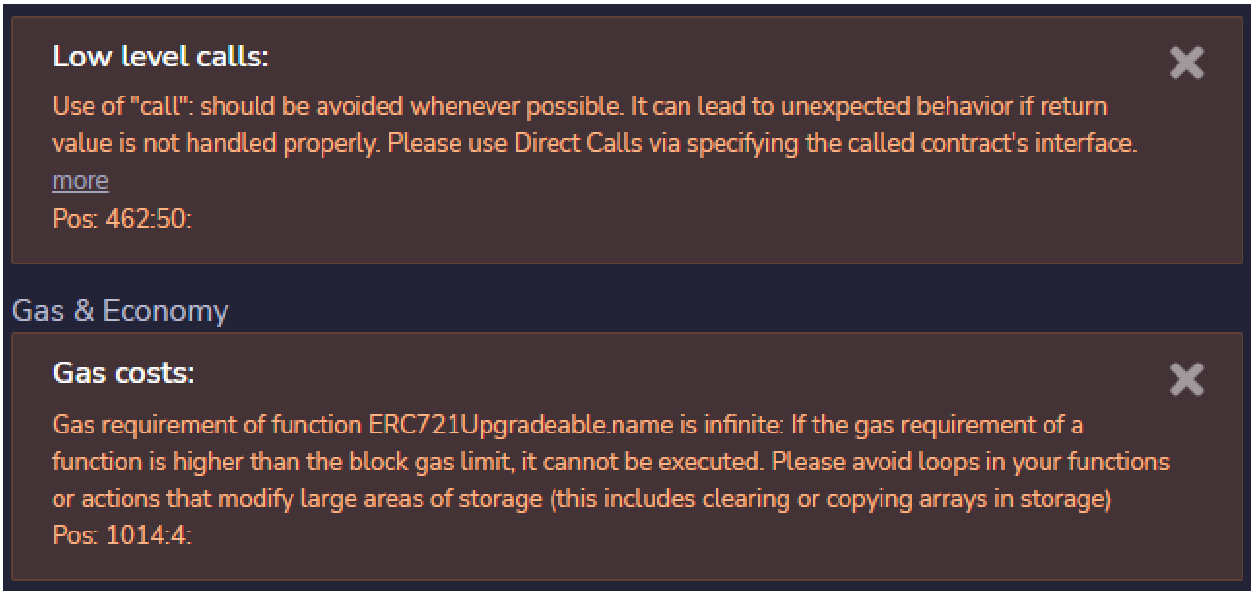


Block timestamp:

Use of "block.timestamp": "block.timestamp" can be influenced by miners to a certain degree. That means that a miner can "choose" the block.timestamp, to a certain degree, to change the outcome of a transaction in the mined block.
more
Pos: 2259:12:

Block timestamp:

Use of "block.timestamp": "block.timestamp" can be influenced by miners to a certain degree. That means that a miner can "choose" the block.timestamp, to a certain degree, to change the outcome of a transaction in the mined block.
more
Pos: 2292:12:



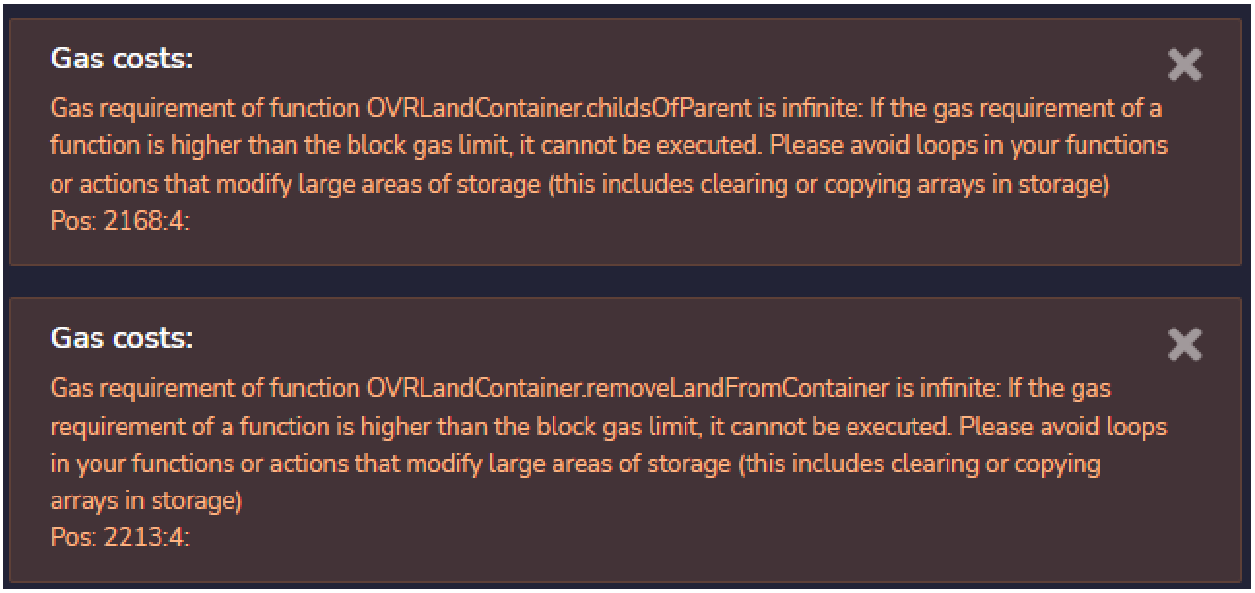
Low level calls:

Use of "call": should be avoided whenever possible. It can lead to unexpected behavior if return value is not handled properly. Please use Direct Calls via specifying the called contract's interface.
more
Pos: 462:50:

Gas & Economy

Gas costs:

Gas requirement of function ERC721Upgradeable.name is infinite: If the gas requirement of a function is higher than the block gas limit, it cannot be executed. Please avoid loops in your functions or actions that modify large areas of storage (this includes clearing or copying arrays in storage)
Pos: 1014:4:

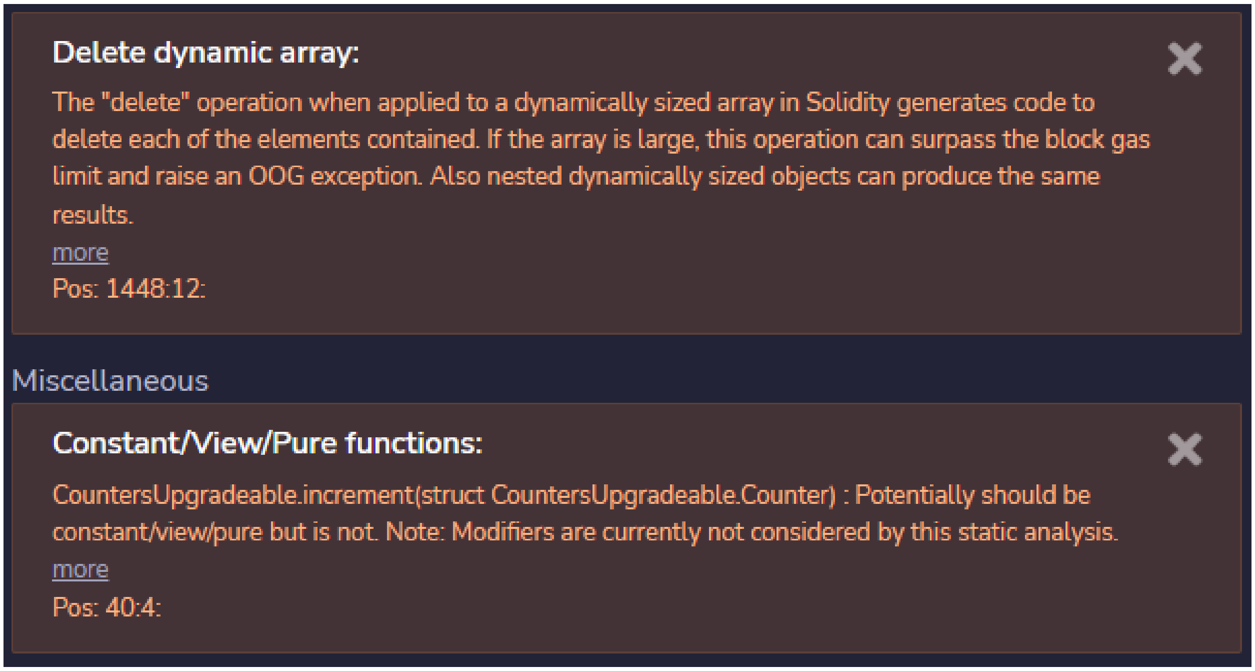


Gas costs:

Gas requirement of function OVRLandContainer.childsOfParent is infinite: If the gas requirement of a function is higher than the block gas limit, it cannot be executed. Please avoid loops in your functions or actions that modify large areas of storage (this includes clearing or copying arrays in storage)
Pos: 2168:4:

Gas costs:

Gas requirement of function OVRLandContainer.removeLandFromContainer is infinite: If the gas requirement of a function is higher than the block gas limit, it cannot be executed. Please avoid loops in your functions or actions that modify large areas of storage (this includes clearing or copying arrays in storage)
Pos: 2213:4:



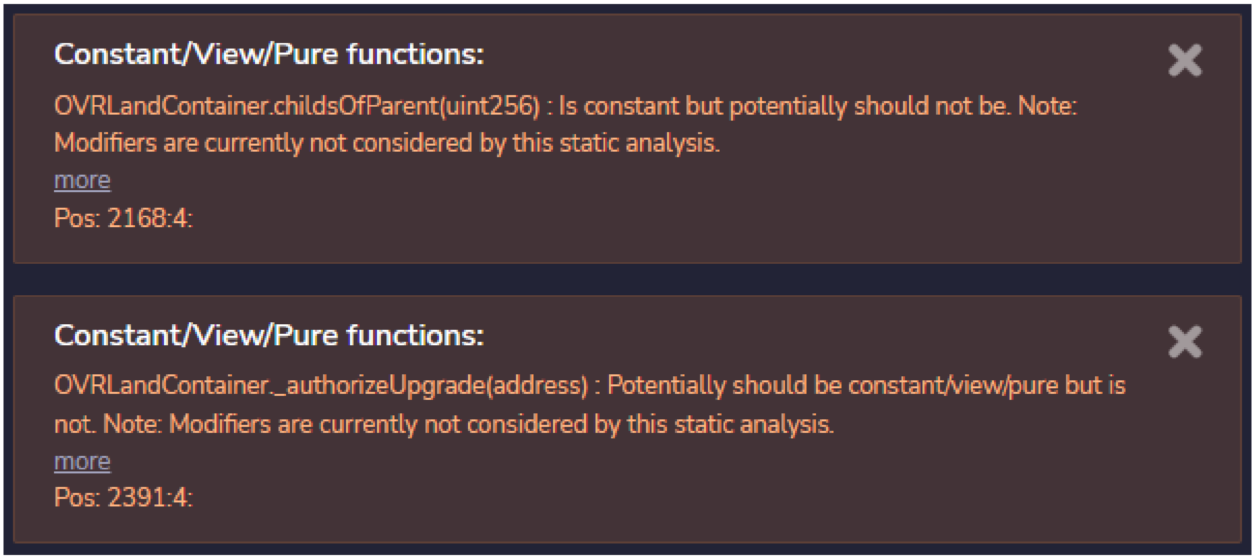
Delete dynamic array:

The "delete" operation when applied to a dynamically sized array in Solidity generates code to delete each of the elements contained. If the array is large, this operation can surpass the block gas limit and raise an OOG exception. Also nested dynamically sized objects can produce the same results.
more
Pos: 1448:12:

Miscellaneous

Constant/View/Pure functions:

CountersUpgradeable.increment(struct CountersUpgradeable.Counter) : Potentially should be constant/view/pure but is not. Note: Modifiers are currently not considered by this static analysis.
more
Pos: 40:4:



Constant/View/Pure functions:

OVRLandContainer.childsOfParent(uint256) : Is constant but potentially should not be. Note: Modifiers are currently not considered by this static analysis.

more

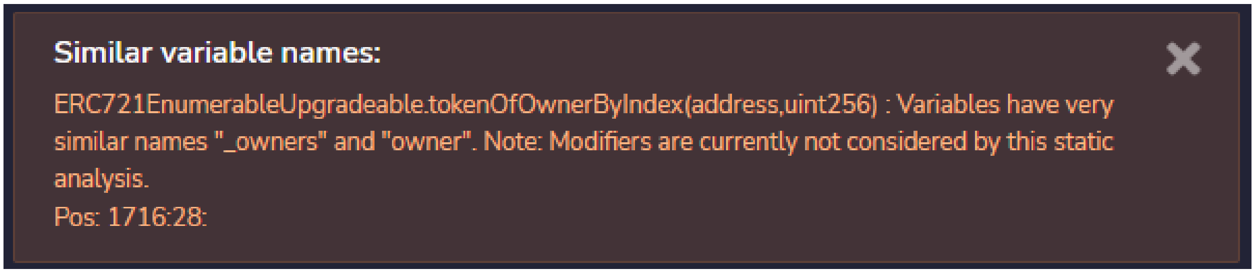
Pos: 2168:4:

Constant/View/Pure functions:

OVRLandContainer._authorizeUpgrade(address) : Potentially should be constant/view/pure but is not. Note: Modifiers are currently not considered by this static analysis.

more

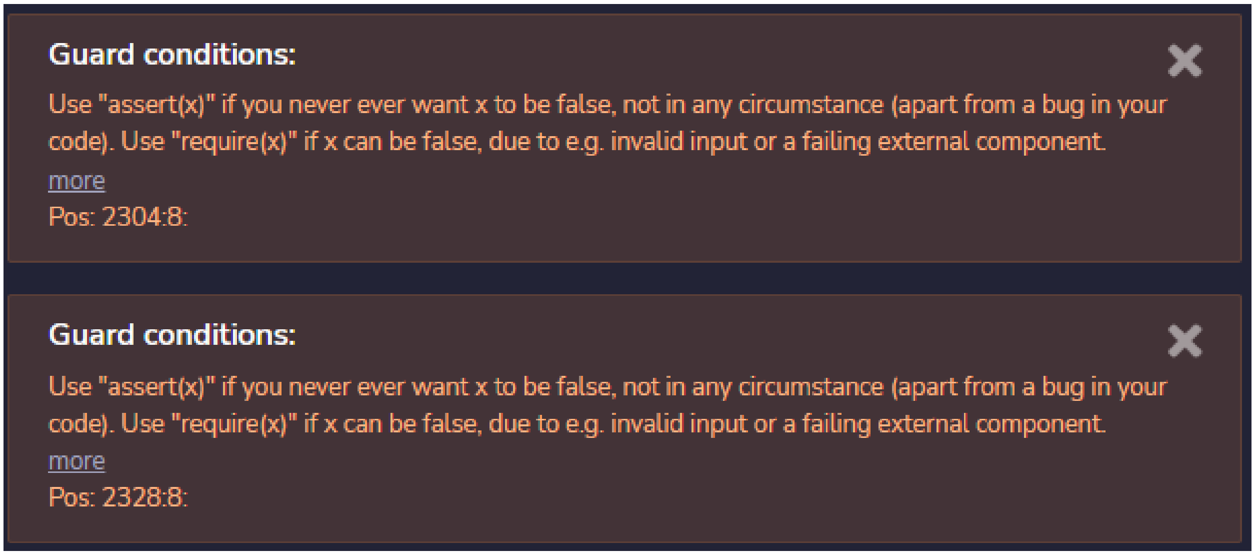
Pos: 2391:4:



Similar variable names:

ERC721EnumerableUpgradeable.tokenOfOwnerByIndex(address,uint256) : Variables have very similar names "_owners" and "owner". Note: Modifiers are currently not considered by this static analysis.

Pos: 1716:28:



Guard conditions:

Use "assert(x)" if you never ever want x to be false, not in any circumstance (apart from a bug in your code). Use "require(x)" if x can be false, due to e.g. invalid input or a failing external component.

more

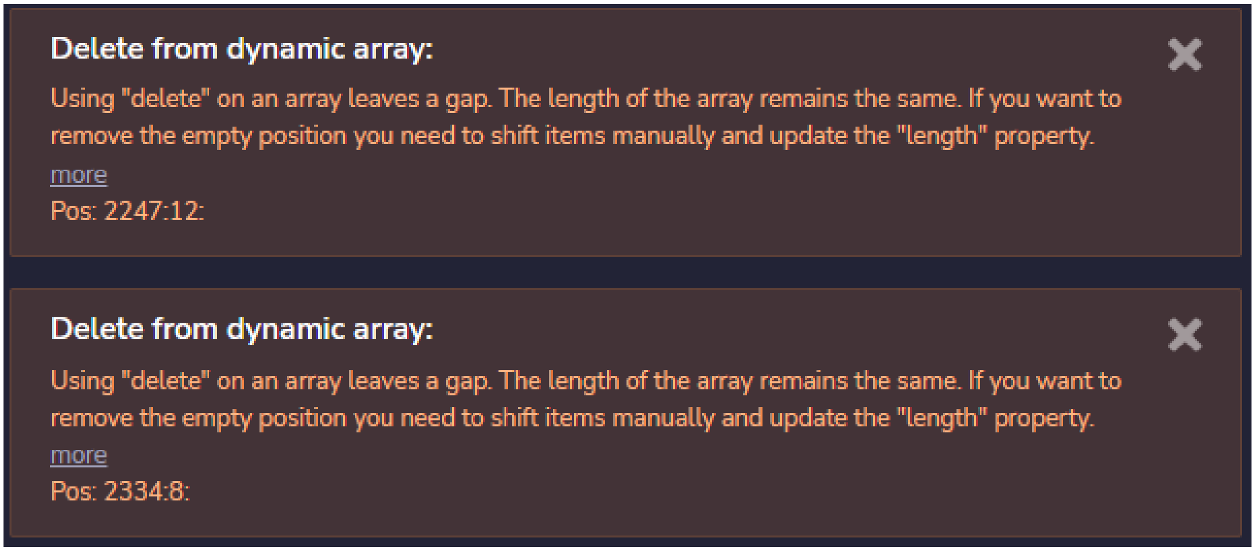
Pos: 2304:8:

Guard conditions:

Use "assert(x)" if you never ever want x to be false, not in any circumstance (apart from a bug in your code). Use "require(x)" if x can be false, due to e.g. invalid input or a failing external component.

more

Pos: 2328:8:



Delete from dynamic array:

Using "delete" on an array leaves a gap. The length of the array remains the same. If you want to remove the empty position you need to shift items manually and update the "length" property.

more

Pos: 2247:12:

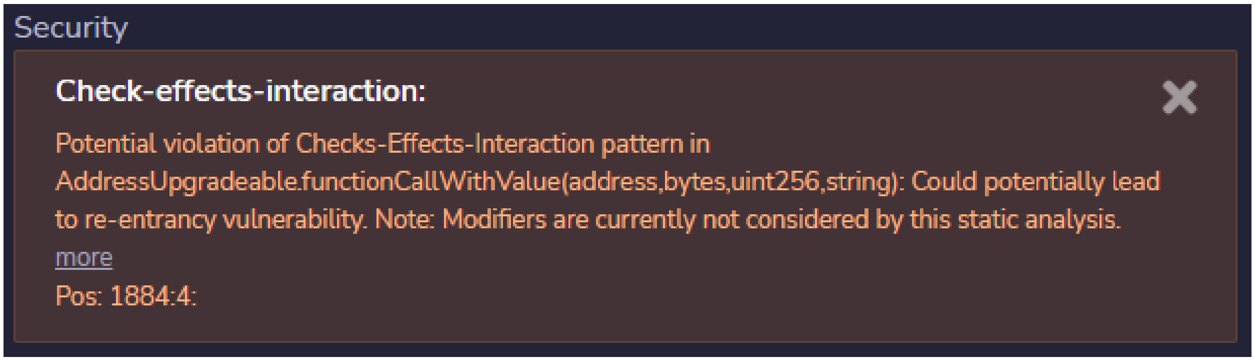
Delete from dynamic array:

Using "delete" on an array leaves a gap. The length of the array remains the same. If you want to remove the empty position you need to shift items manually and update the "length" property.

more

Pos: 2334:8:

OVRMarketplace.sol



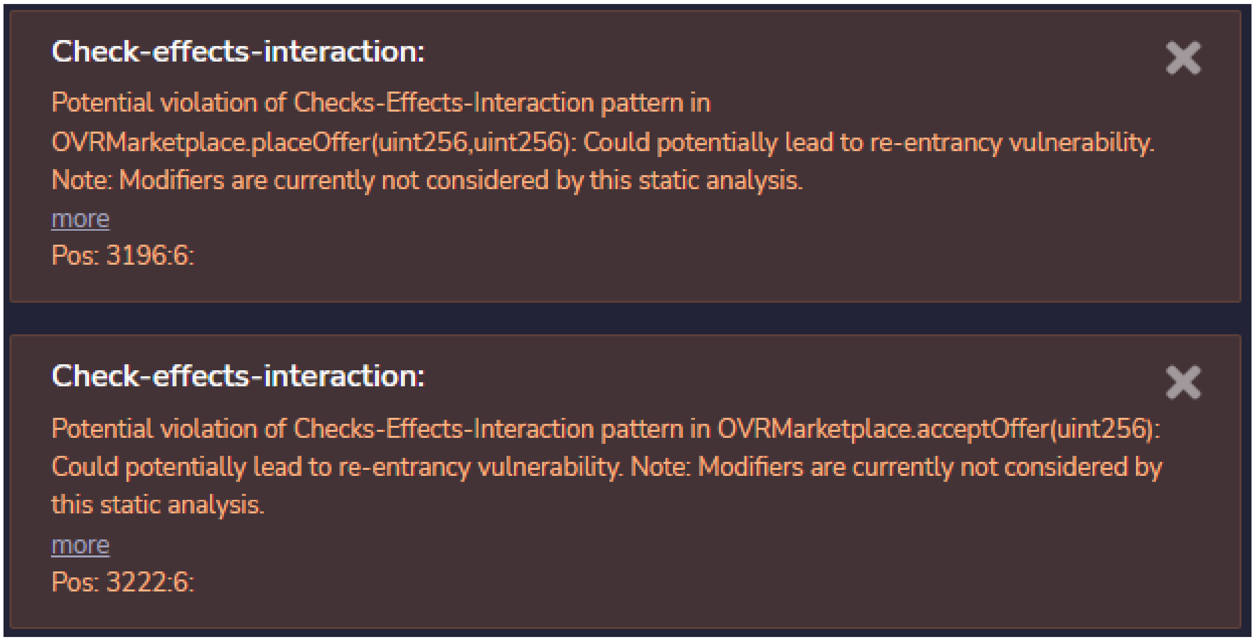
Security

Check-effects-interaction:

Potential violation of Checks-Effects-Interaction pattern in AddressUpgradeable.functionCallWithValue(address,bytes,uint256,string): Could potentially lead to re-entrancy vulnerability. Note: Modifiers are currently not considered by this static analysis.

more

Pos: 1884:4:



Check-effects-interaction:

Potential violation of Checks-Effects-Interaction pattern in OVRMarketplace.placeOffer(uint256,uint256): Could potentially lead to re-entrancy vulnerability. Note: Modifiers are currently not considered by this static analysis.

more

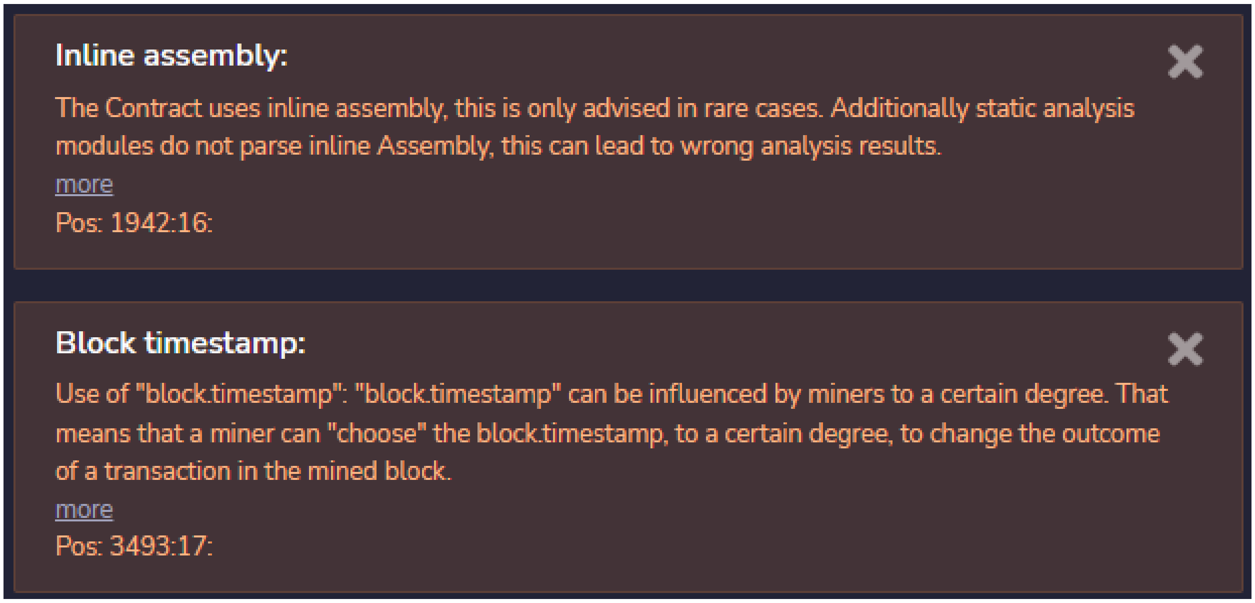
Pos: 3196:6:

Check-effects-interaction:

Potential violation of Checks-Effects-Interaction pattern in OVRMarketplace.acceptOffer(uint256): Could potentially lead to re-entrancy vulnerability. Note: Modifiers are currently not considered by this static analysis.

more

Pos: 3222:6:



Inline assembly:

The Contract uses inline assembly, this is only advised in rare cases. Additionally static analysis modules do not parse inline Assembly, this can lead to wrong analysis results.

more

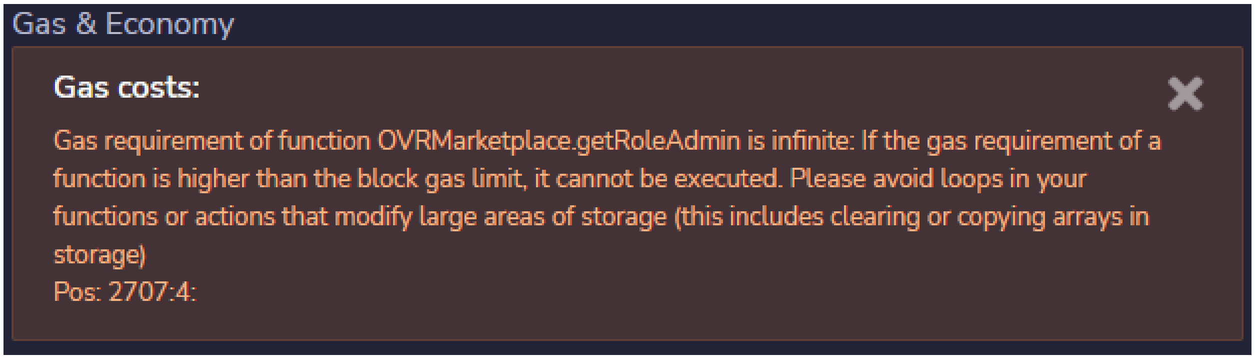
Pos: 1942:16:

Block timestamp:

Use of "block.timestamp": "block.timestamp" can be influenced by miners to a certain degree. That means that a miner can "choose" the block.timestamp, to a certain degree, to change the outcome of a transaction in the mined block.

more

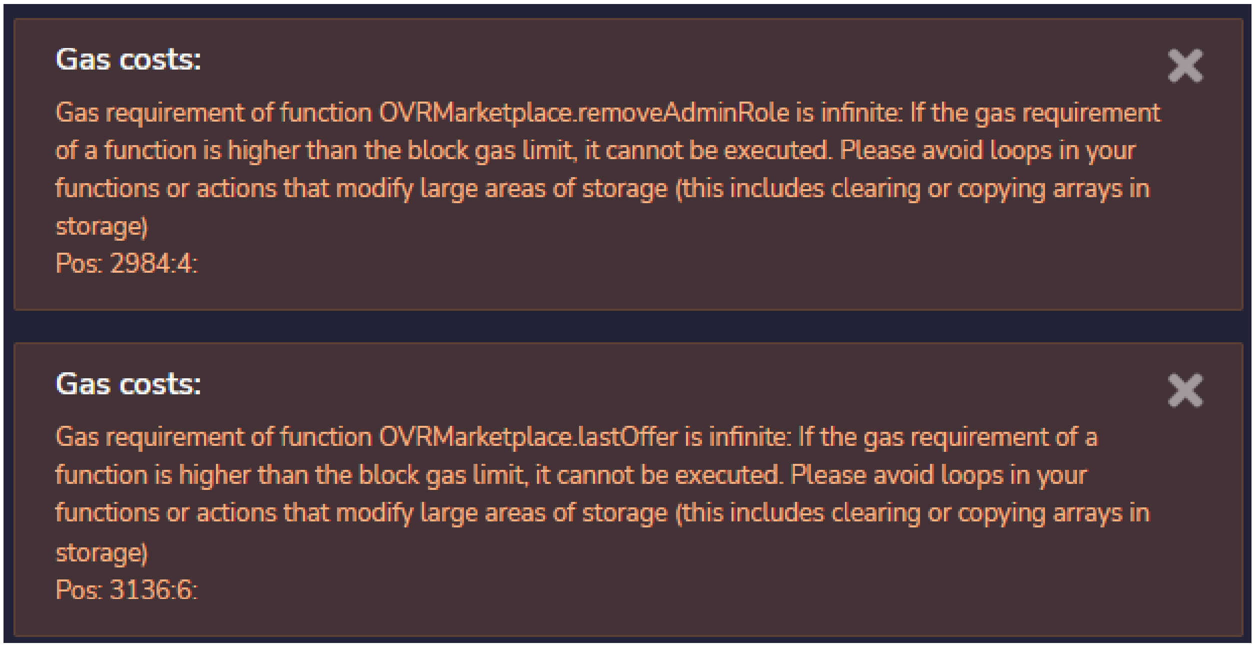
Pos: 3493:17:



Gas & Economy

Gas costs:

Gas requirement of function OVRMarketplace.getRoleAdmin is infinite: If the gas requirement of a function is higher than the block gas limit, it cannot be executed. Please avoid loops in your functions or actions that modify large areas of storage (this includes clearing or copying arrays in storage)
Pos: 2707:4:

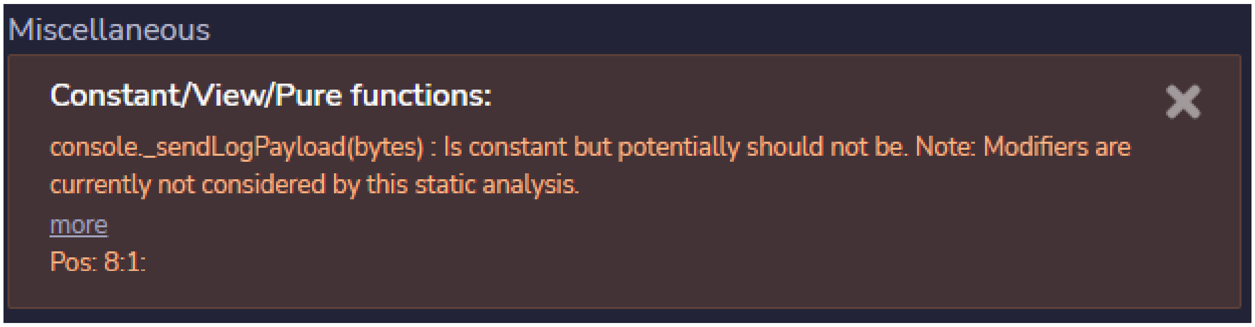


Gas costs:

Gas requirement of function OVRMarketplace.removeAdminRole is infinite: If the gas requirement of a function is higher than the block gas limit, it cannot be executed. Please avoid loops in your functions or actions that modify large areas of storage (this includes clearing or copying arrays in storage)
Pos: 2984:4:

Gas costs:

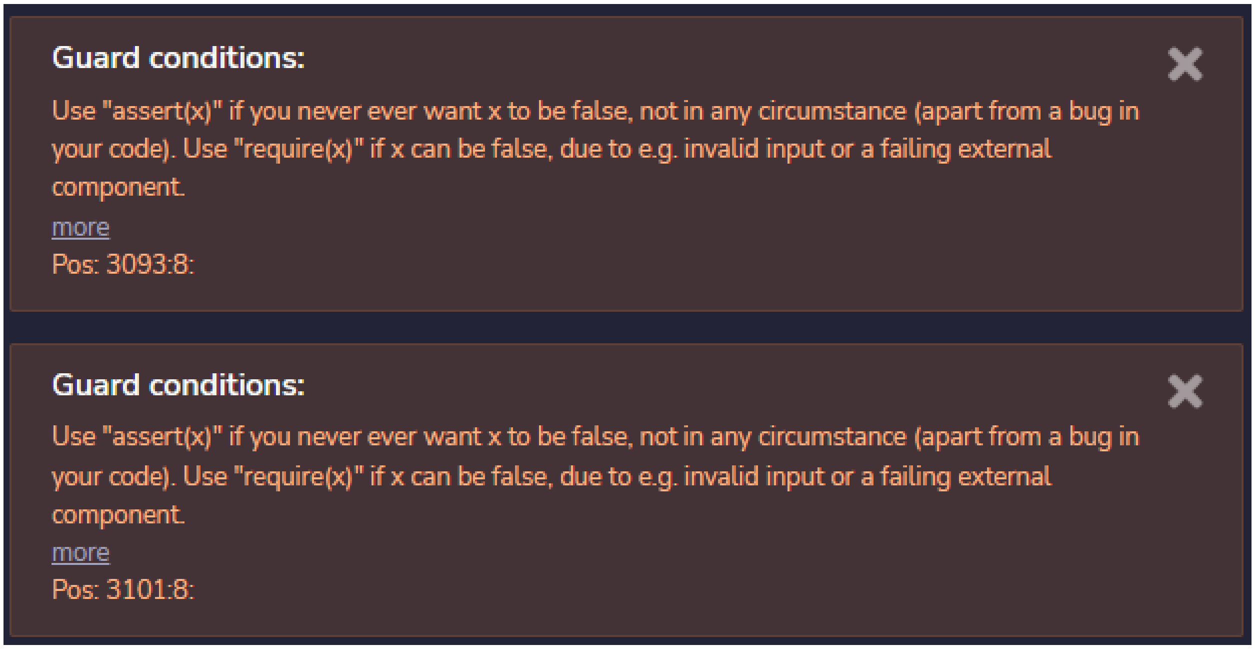
Gas requirement of function OVRMarketplace.lastOffer is infinite: If the gas requirement of a function is higher than the block gas limit, it cannot be executed. Please avoid loops in your functions or actions that modify large areas of storage (this includes clearing or copying arrays in storage)
Pos: 3136:6:



Miscellaneous

Constant/View/Pure functions:

console._sendLogPayload(bytes) : Is constant but potentially should not be. Note: Modifiers are currently not considered by this static analysis.
more
Pos: 8:1:



Guard conditions:

Use "assert(x)" if you never ever want x to be false, not in any circumstance (apart from a bug in your code). Use "require(x)" if x can be false, due to e.g. invalid input or a failing external component.

more

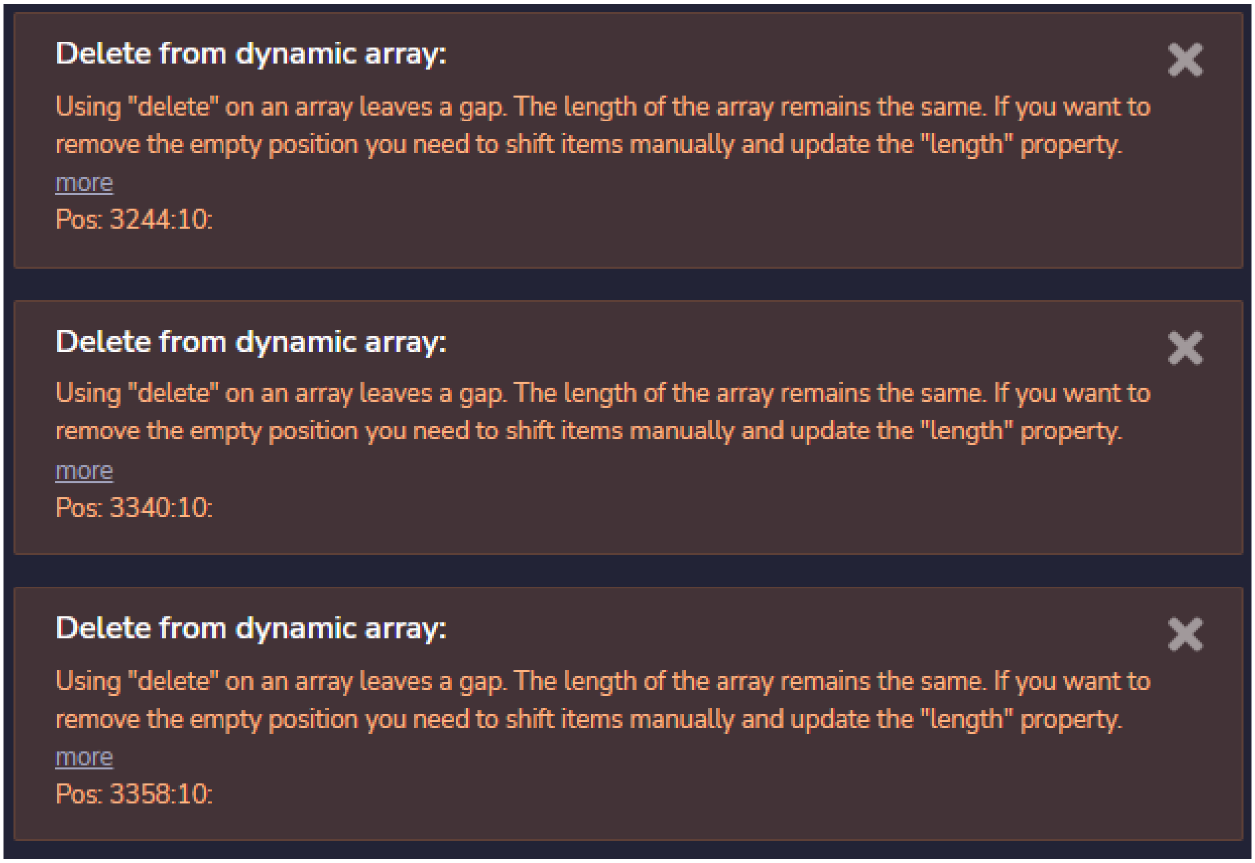
Pos: 3093:8:

Guard conditions:

Use "assert(x)" if you never ever want x to be false, not in any circumstance (apart from a bug in your code). Use "require(x)" if x can be false, due to e.g. invalid input or a failing external component.

more

Pos: 3101:8:



Delete from dynamic array:

Using "delete" on an array leaves a gap. The length of the array remains the same. If you want to remove the empty position you need to shift items manually and update the "length" property.
more
Pos: 3244:10:

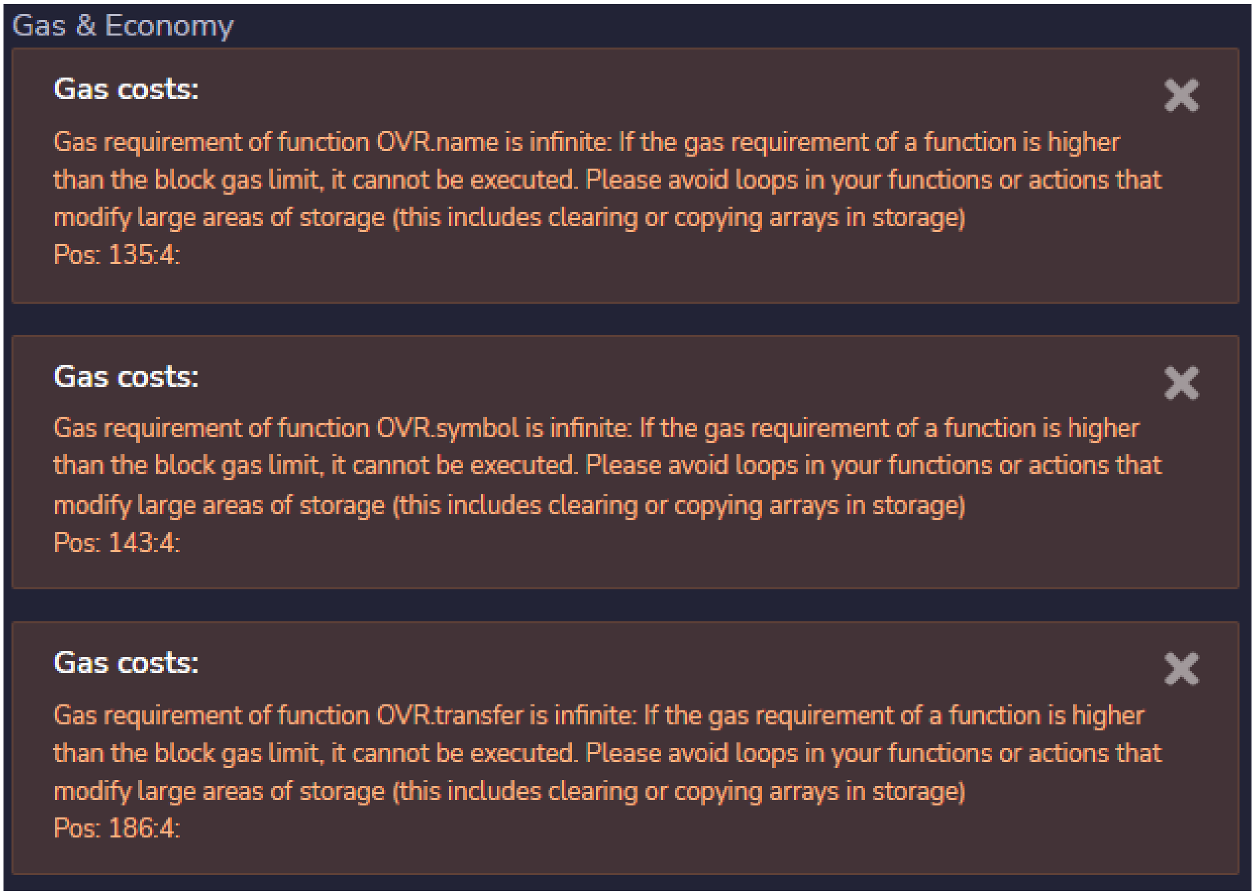
Delete from dynamic array:

Using "delete" on an array leaves a gap. The length of the array remains the same. If you want to remove the empty position you need to shift items manually and update the "length" property.
more
Pos: 3340:10:

Delete from dynamic array:

Using "delete" on an array leaves a gap. The length of the array remains the same. If you want to remove the empty position you need to shift items manually and update the "length" property.
more
Pos: 3358:10:

OVRToken.sol



Gas & Economy

Gas costs:

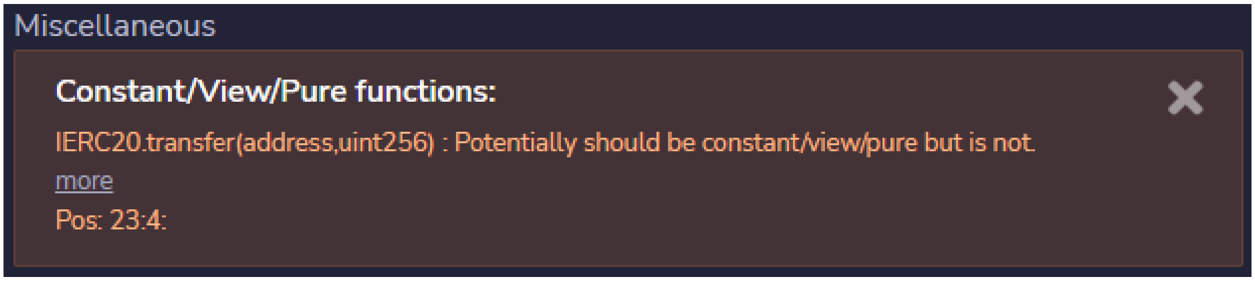
Gas requirement of function OVR.name is infinite: If the gas requirement of a function is higher than the block gas limit, it cannot be executed. Please avoid loops in your functions or actions that modify large areas of storage (this includes clearing or copying arrays in storage)
Pos: 135:4:

Gas costs:

Gas requirement of function OVR.symbol is infinite: If the gas requirement of a function is higher than the block gas limit, it cannot be executed. Please avoid loops in your functions or actions that modify large areas of storage (this includes clearing or copying arrays in storage)
Pos: 143:4:

Gas costs:

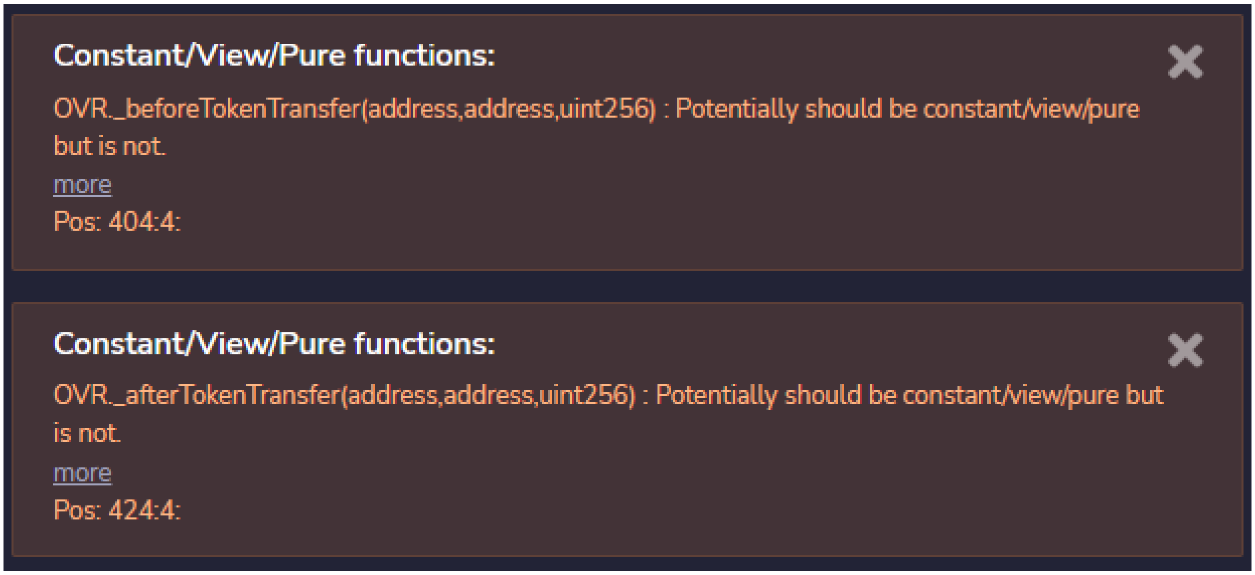
Gas requirement of function OVR.transfer is infinite: If the gas requirement of a function is higher than the block gas limit, it cannot be executed. Please avoid loops in your functions or actions that modify large areas of storage (this includes clearing or copying arrays in storage)
Pos: 186:4:



Miscellaneous

Constant/View/Pure functions:

IERC20.transfer(address,uint256) : Potentially should be constant/view/pure but is not.
more
Pos: 23:4:



Constant/View/Pure functions:

OVR._beforeTokenTransfer(address,address,uint256) : Potentially should be constant/view/pure but is not.

more

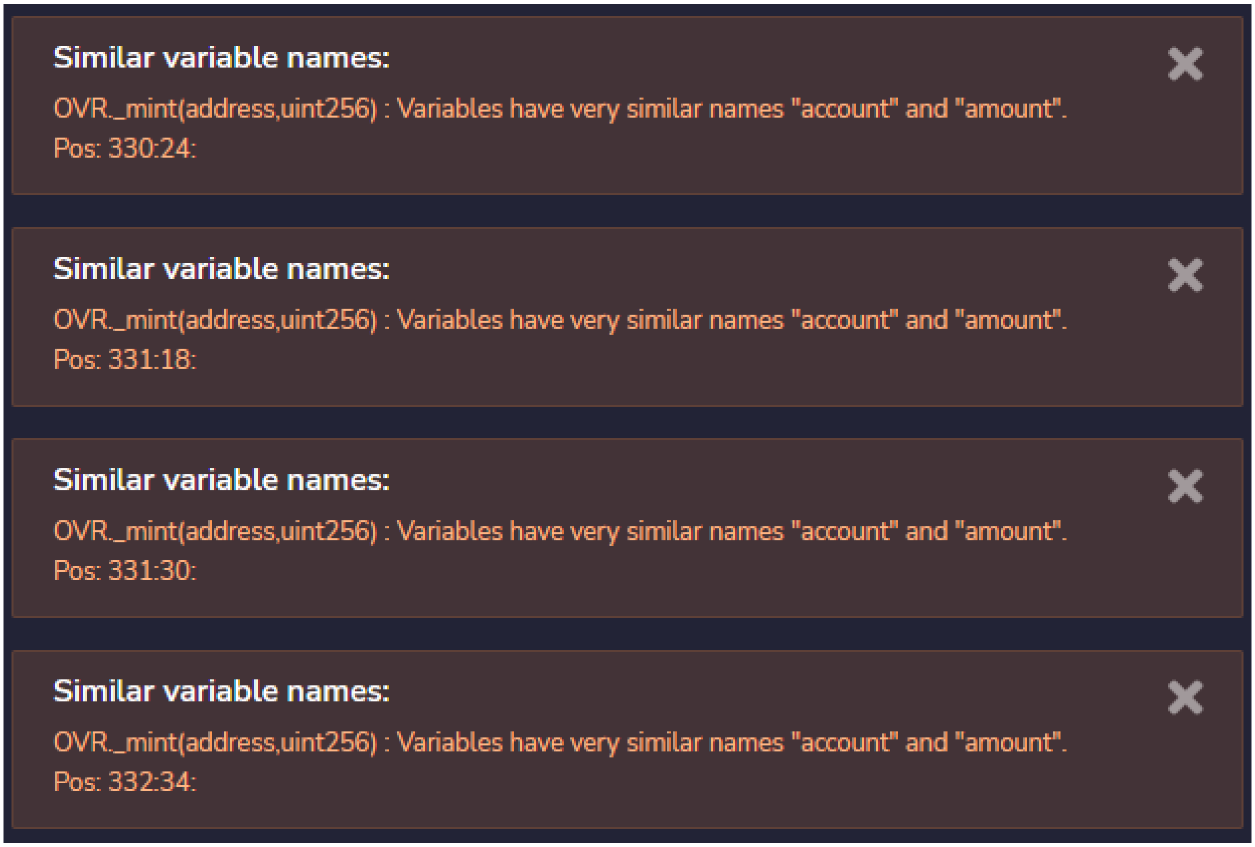
Pos: 404:4:

Constant/View/Pure functions:

OVR._afterTokenTransfer(address,address,uint256) : Potentially should be constant/view/pure but is not.

more

Pos: 424:4:



Similar variable names:

OVR._mint(address,uint256) : Variables have very similar names "account" and "amount".

Pos: 330:24:

Similar variable names:

OVR._mint(address,uint256) : Variables have very similar names "account" and "amount".

Pos: 331:18:

Similar variable names:

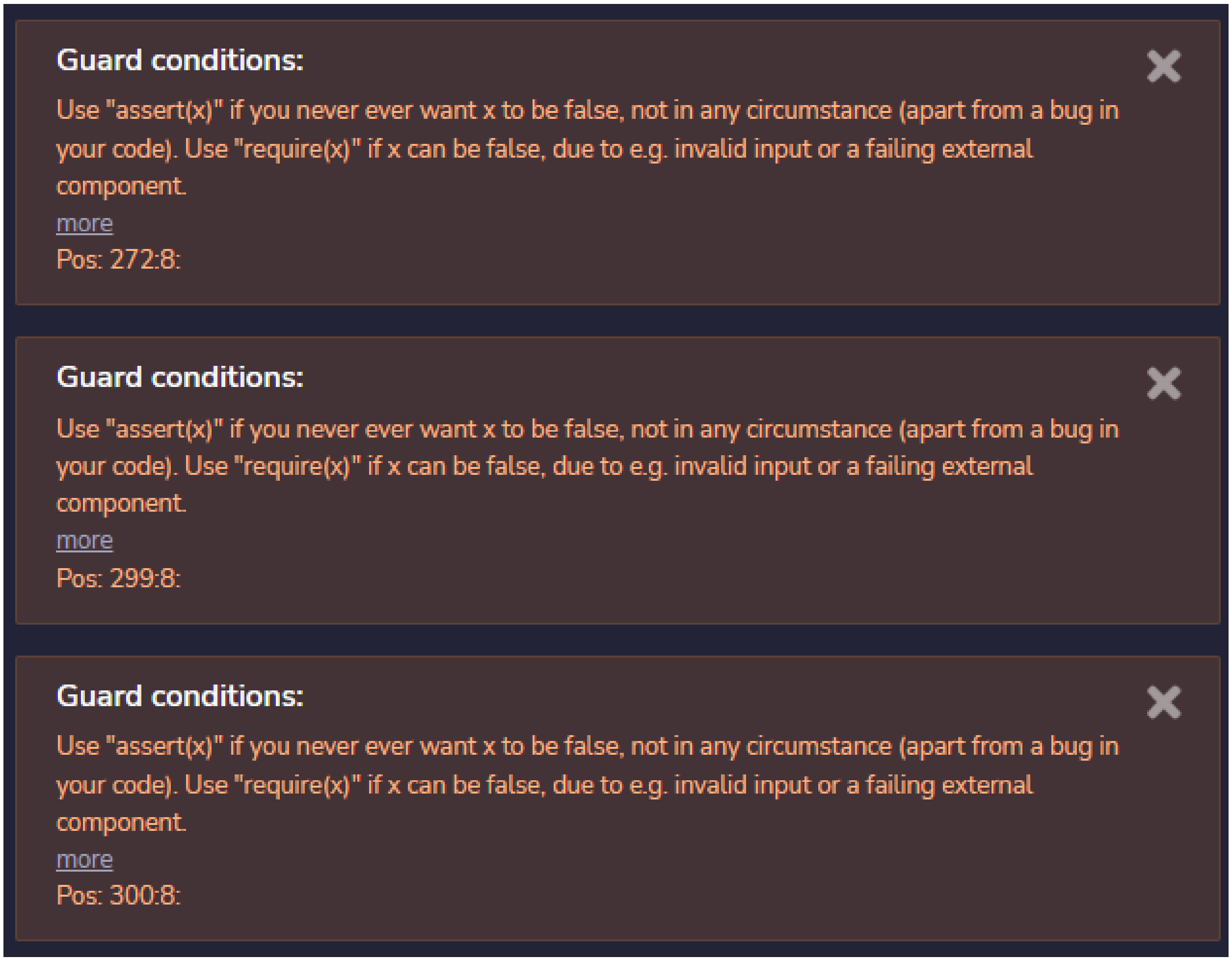
OVR._mint(address,uint256) : Variables have very similar names "account" and "amount".

Pos: 331:30:

Similar variable names:

OVR._mint(address,uint256) : Variables have very similar names "account" and "amount".

Pos: 332:34:



Guard conditions:

Use "assert(x)" if you never ever want x to be false, not in any circumstance (apart from a bug in your code). Use "require(x)" if x can be false, due to e.g. invalid input or a failing external component.

more

Pos: 272:8:

Guard conditions:

Use "assert(x)" if you never ever want x to be false, not in any circumstance (apart from a bug in your code). Use "require(x)" if x can be false, due to e.g. invalid input or a failing external component.

more

Pos: 299:8:

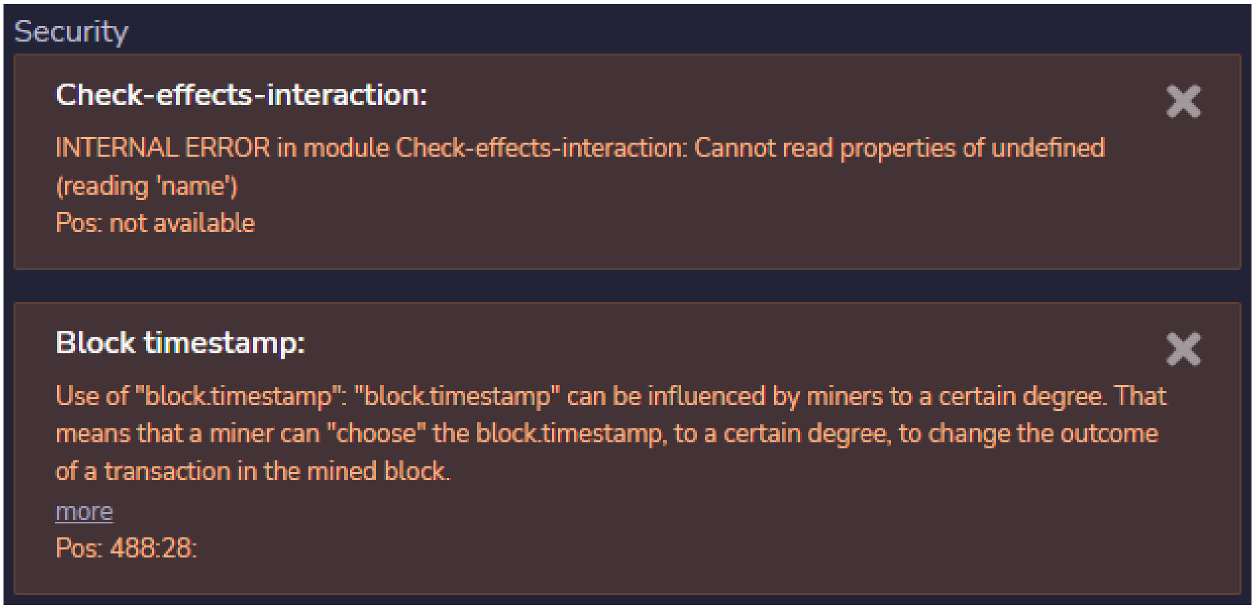
Guard conditions:

Use "assert(x)" if you never ever want x to be false, not in any circumstance (apart from a bug in your code). Use "require(x)" if x can be false, due to e.g. invalid input or a failing external component.

more

Pos: 300:8:

Uniswapv2router.sol



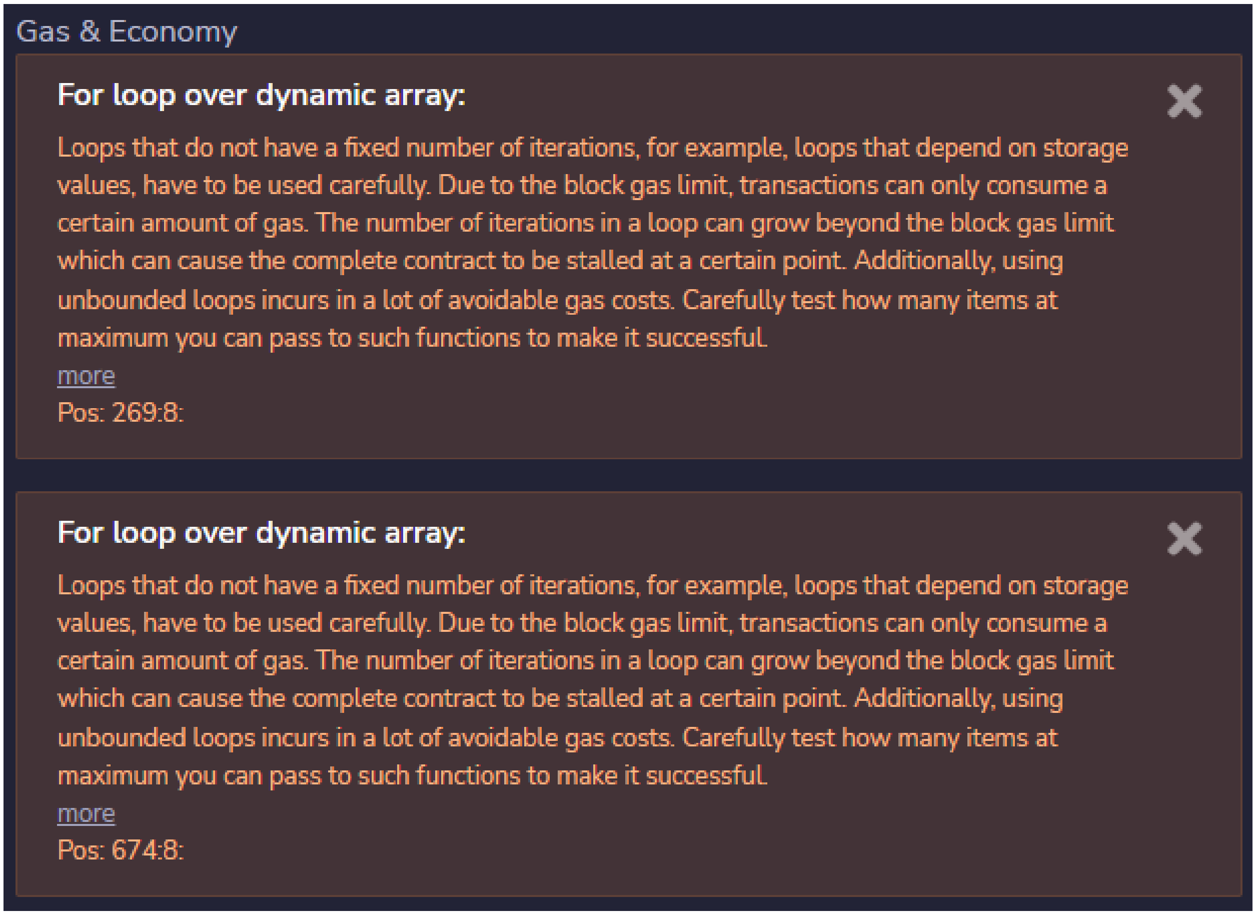
Security

Check-effects-interaction:

INTERNAL ERROR in module Check-effects-interaction: Cannot read properties of undefined (reading 'name')
Pos: not available

Block timestamp:

Use of "block.timestamp": "block.timestamp" can be influenced by miners to a certain degree. That means that a miner can "choose" the block.timestamp, to a certain degree, to change the outcome of a transaction in the mined block.
more
Pos: 488:28:



Gas & Economy

For loop over dynamic array:

Loops that do not have a fixed number of iterations, for example, loops that depend on storage values, have to be used carefully. Due to the block gas limit, transactions can only consume a certain amount of gas. The number of iterations in a loop can grow beyond the block gas limit which can cause the complete contract to be stalled at a certain point. Additionally, using unbounded loops incurs in a lot of avoidable gas costs. Carefully test how many items at maximum you can pass to such functions to make it successful.

more

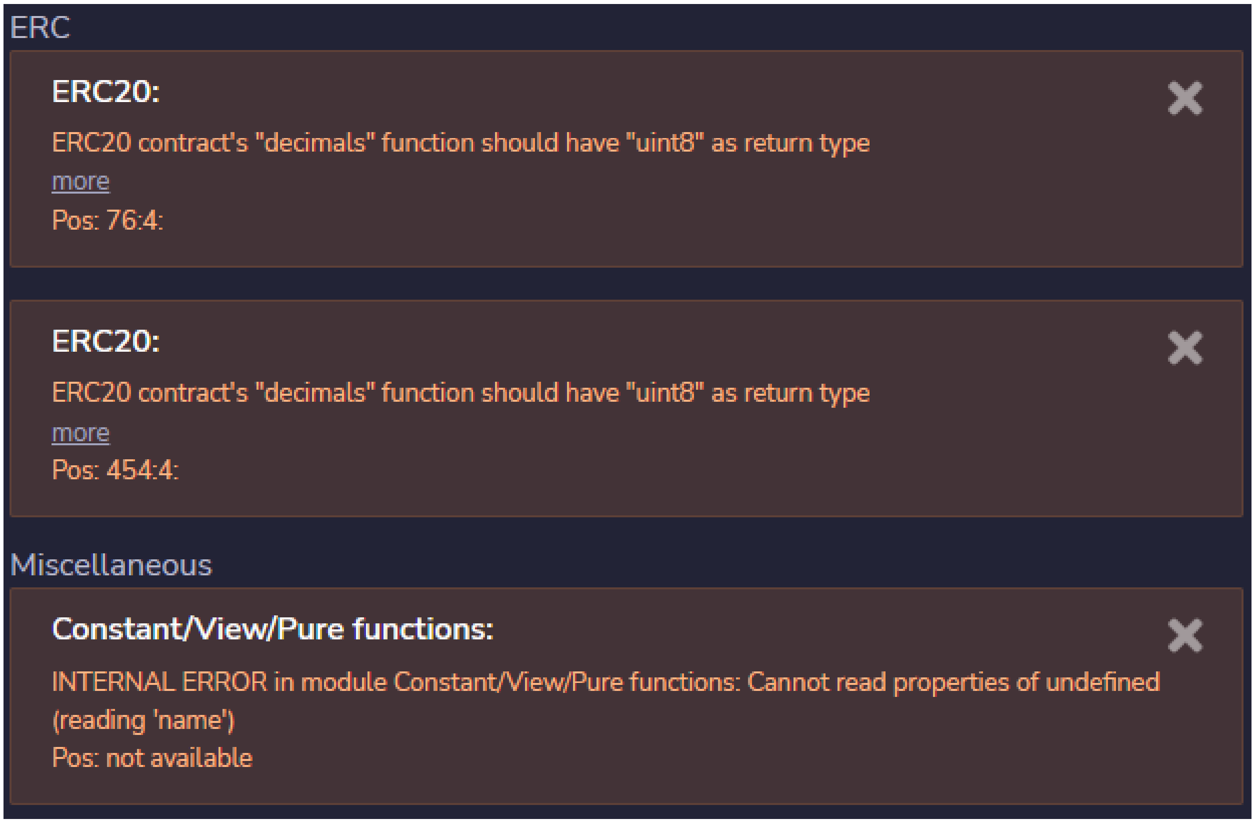
Pos: 269:8:

For loop over dynamic array:

Loops that do not have a fixed number of iterations, for example, loops that depend on storage values, have to be used carefully. Due to the block gas limit, transactions can only consume a certain amount of gas. The number of iterations in a loop can grow beyond the block gas limit which can cause the complete contract to be stalled at a certain point. Additionally, using unbounded loops incurs in a lot of avoidable gas costs. Carefully test how many items at maximum you can pass to such functions to make it successful.

more

Pos: 674:8:



ERC

ERC20:

ERC20 contract's "decimals" function should have "uint8" as return type

more

Pos: 76:4:

ERC20:

ERC20 contract's "decimals" function should have "uint8" as return type

more

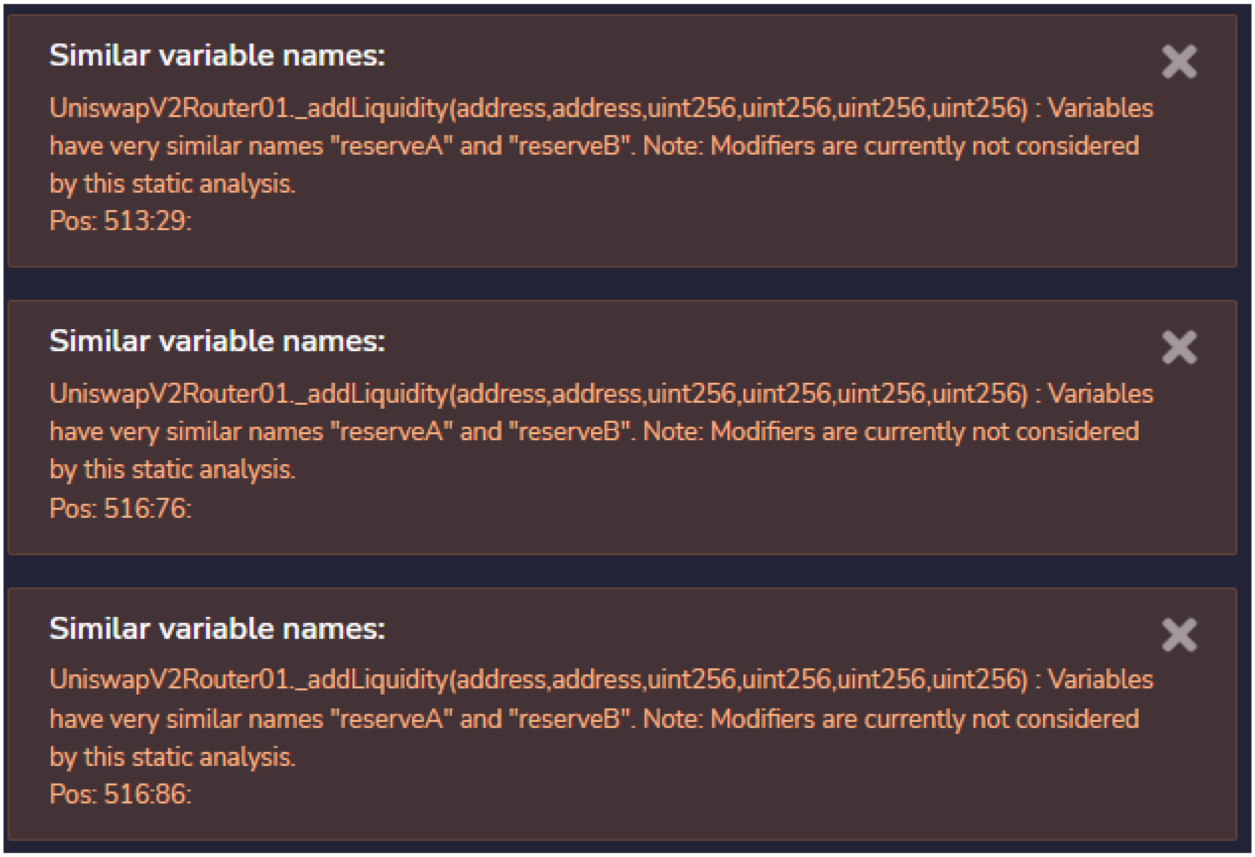
Pos: 454:4:

Miscellaneous

Constant/View/Pure functions:

INTERNAL ERROR in module Constant/View/Pure functions: Cannot read properties of undefined (reading 'name')

Pos: not available



Similar variable names:

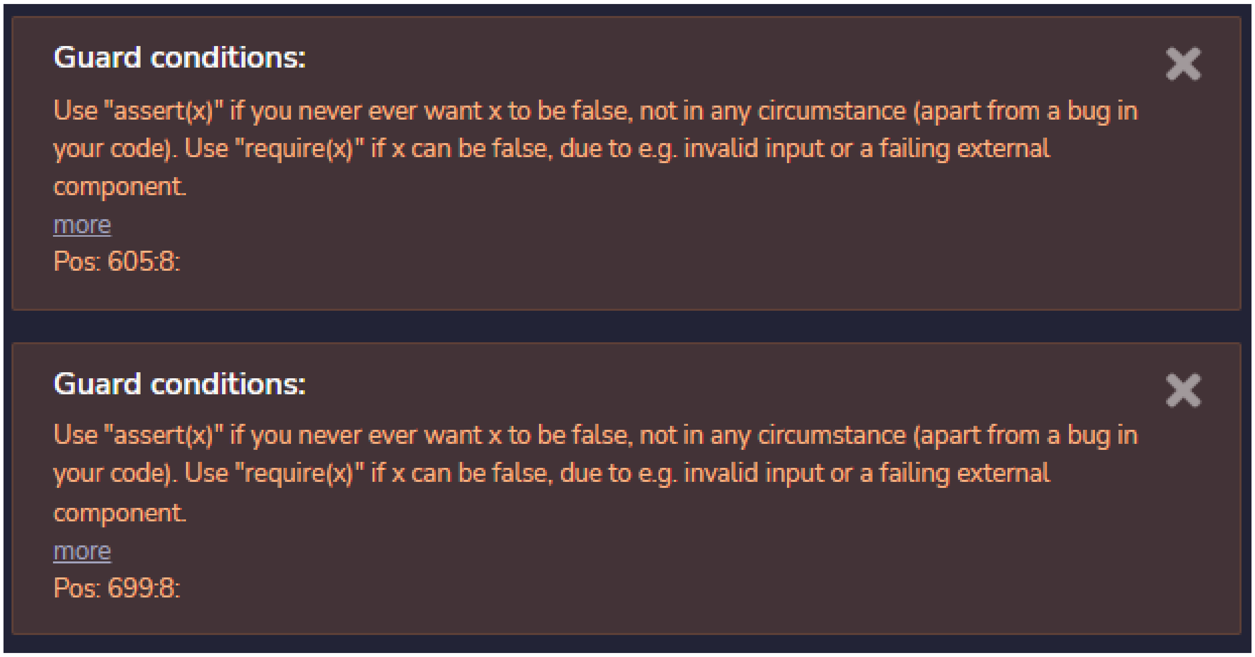
UniswapV2Router01._addLiquidity(address,address,uint256,uint256,uint256,uint256) : Variables have very similar names "reserveA" and "reserveB". Note: Modifiers are currently not considered by this static analysis.
Pos: 513:29:

Similar variable names:

UniswapV2Router01._addLiquidity(address,address,uint256,uint256,uint256,uint256) : Variables have very similar names "reserveA" and "reserveB". Note: Modifiers are currently not considered by this static analysis.
Pos: 516:76:

Similar variable names:

UniswapV2Router01._addLiquidity(address,address,uint256,uint256,uint256,uint256) : Variables have very similar names "reserveA" and "reserveB". Note: Modifiers are currently not considered by this static analysis.
Pos: 516:86:



Guard conditions:

Use "assert(x)" if you never ever want x to be false, not in any circumstance (apart from a bug in your code). Use "require(x)" if x can be false, due to e.g. invalid input or a failing external component.

more

Pos: 605:8:

Guard conditions:

Use "assert(x)" if you never ever want x to be false, not in any circumstance (apart from a bug in your code). Use "require(x)" if x can be false, due to e.g. invalid input or a failing external component.

more

Pos: 699:8:

Severity Definitions

Risk Level	Description
Critical	Critical vulnerabilities are usually straightforward to exploit and can lead to lost tokens etc.
High	High level vulnerabilities are difficult to exploit; however, they also have a significant impact on smart contract execution, e.g. public access to crucial functions.
Medium	Medium level vulnerabilities are important to fix; however, they cannot lead to lost tokens.
Low	Low level vulnerabilities are most related to outdated, unused etc. These code snippets cannot have a significant impact on execution.
Lowest Code Style/ Best Practice	Lowest level vulnerabilities, code style violations and information statements cannot affect smart contract execution and can be ignored.

Audit Findings

Critical:

No critical severity vulnerabilities were found.

High:

No high severity vulnerabilities were found.

Medium:

No medium severity vulnerabilities were found.

Low:

No low severity vulnerabilities were found.

Very Low:

No very low severity vulnerabilities were found.

Conclusion

We were given a contract file and have used all possible tests based on the given object. The contract is written systematically, so it is now ready to go for production.

We have used all the latest static tools and manual observations to cover maximum possible test cases to scan everything.

The security state of the reviewed contract is now “well secured”

Note For Contract Users

Owner has full control over the smart contract. Thus, technical auditing does not guarantee the project's ethical side.

Please do your due diligence before investing. Our audit report is never an investment advice.

Our Methodology

We like to work with a transparent process and make our reviews a collaborative effort. The goals of our security audits are to improve the quality of systems we review and aim for sufficient remediation to help protect users. The following is the methodology we use in our security audit process.

Manual Code Review

In manually reviewing all of the code, we look for any potential issues with code logic, error handling, protocol and header parsing, cryptographic errors, and random number generators. We also watch for areas where more defensive programming could reduce the risk of future mistakes and speed up future audits. Although our primary focus is on the in-scope code, we examine dependency code and behavior when it is relevant to a particular line of investigation.

Vulnerability Analysis

Our audit techniques included manual code analysis, user interface interaction, and whitebox penetration testing. We look at the project's web site to get a high level understanding of what functionality the software under review provides. We then meet with the developers to gain an appreciation of their vision of the software. We install and use the relevant software, exploring the user interactions and roles. While we do this, we brainstorm threat models and attack surfaces. We read design documentation, review other audit results, search for similar projects, examine source code dependencies, skim open issue tickets, and generally investigate details other than the implementation.

Documenting Results

We follow a conservative, transparent process for analyzing potential security vulnerabilities and seeing them through successful remediation. Whenever a potential issue is discovered, we immediately create an Issue entry for it in this document, even though we have not yet verified the feasibility and impact of the issue. This process is conservative because we document our suspicions early even if they are later shown to not represent exploitable vulnerabilities. We generally follow a process of first documenting the suspicion with unresolved questions, then confirming the issue through code analysis, live experimentation, or automated tests. Code analysis is the most tentative, and we strive to provide test code, log captures, or screenshots demonstrating our confirmation. After this we analyse the feasibility of an attack in a live system.

Suggested Solutions

We search for immediate mitigations that live deployments can take, and finally we suggest the requirements for remediation engineering for future releases. The mitigation and remediation recommendations should be scrutinised by the developers and deployment engineers, and successful mitigation and remediation is an ongoing collaborative process after we deliver our report, and before the details are made public.

Disclaimers

RD Auditors Disclaimer

The smart contracts given for audit have been analysed in accordance with the best industry practices at the date of this report, in relation to: cybersecurity vulnerabilities and issues in smart contract source code, the details of which are disclosed in this report, (Source Code); the Source Code compilation, deployment and functionality (performing the intended functions).

Because the total number of test cases are unlimited, the audit makes no statements or warranties on the security of the code. It also cannot be considered as a sufficient assessment regarding the utility and safety of the code, bugfree status or any other statements of the contract. While we have done our best in conducting the analysis and producing this report, it is important to note that you should not rely on this report only - we recommend proceeding with several independent audits and a public bug bounty program to ensure security of smart contracts.

Technical Disclaimer

Smart contracts are deployed and executed on the blockchain. The platform, its programming language, and other software related to the smart contract can have their own vulnerabilities that can lead to hacks. Thus, the audit can’t guarantee explicit security of the audited smart contracts.

RD
AUDITORS

Email: info@rdauditors.com
Website: www.rdauditors.com