// HALBORN

BSCEX LAUNCHPADx

Smart Contract Security Audit

Prepared by: **Halborn**

Date of Engagement: **February 25, 2021**

Visit: **Halborn.com**

DOCUMENT REVISION HISTORY

VERSION	MODIFICATION	DATE	AUTHOR
0.1	Document Creation	02/25/2021	Nishit Majithia

CONTACTS

CONTACT	COMPANY	EMAIL
Rob Behnke	Halborn	Rob.Behnke@halborn.com
Steven Walbroehl	Halborn	Steven.Walbroehl@halborn.com
Nishit Majithia	Halborn	nishit.majithia@halborn.com

EXECUTIVE SUMMARY

1.1 INTRODUCTION

BSCEX is a decentralized non-custodial cryptocurrency exchange that aims to facilitate the services that exchange-centered ecosystems provide. This project's primary goal is to bring Binance's off-chain services on-chain to the Binance Smart Chain (BSC). The security assessment was scoped to the smart contract LaunchpadxTicket.sol, LaunchpadxFactory.sol and LaunchpadxIDO.sol. An audit of the security risk and implications regarding the changes introduced by the development team at BSCEX prior to its production release shortly following the assessments deadline.

Though the outcome of this security audit is satisfactory; due to time and resource constraints, only testing and verication of essential properties were performed to achieve objectives and deliverables set in the scope. It is important to remark the use of the best practices for secure smart contract development. Halborn recommends performing further testing to validate extended safety and correctness in context to the whole set of contracts. External threats, such as economic attacks, oracle attacks, and inter-contract functions and calls should be validated for expected logic and state.

1.2 TEST APPROACH & METHODOLOGY

Halborn performed a combination of manual and automated security testing to balance efficiency, timeliness, practicality, and accuracy in regard to the scope of the smart contract audit.
While manual testing is recommended to uncover flaws in logic, process, and implementation; automated testing techniques help enhance coverage of smart contracts and can quickly identify items that do not follow security best practices. The following phases and associated tools were used throughout the term of the audit:
- Research into architecture and purpose
- Smart Contract manual code read and walkthrough
- Graphing out functionality and contract logic/connectivity/functions (solgraph)

- Manual Assessment of use and safety for the critical solidity variables and functions in scope to identify any arithmetic related vulnerability classes.
- Scanning of solidity files for vulnerabilities, security hotspots, or bugs. (MythX)
- Static Analysis of security for scoped contract, and imported functions. (Slither)
- Testnet deployment (Truffle, Ganache, Infura)
- Smart Contract Fuzzing and dynamic state exploitation (Echidna) Symbolic Execution / EVM bytecode security assessment (limited time)

1.3 SCOPE

IN-SCOPE:
- LaunchpadxTicket.sol
- LaunchpadxFactory.sol
- LaunchpadxIDO.sol

Specific commit of contract: commit
38d61d0760ac35f926572600aa687d4f8778dc28

OUT-OF-SCOPE:
Other smart contracts in the repository, external libraries and economics attacks.

2. ASSESSMENT SUMMARY & FINDINGS OVERVIEW

CRITICAL	HIGH	MEDIUM	LOW	INFORMATIONAL
0	0	0	1	2

SECURITY ANALYSIS	RISK LEVEL	REMEDIATION DATE
MISSING ADDRESS CHECK	Low	-
IGNORE RETURN VALUES	Informational	-
STATIC ANALYSIS	Informational	-

FINDINGS & TECH DETAILS

3.1 MISSING ADDRESS CHECK - LOW

Description:

Address validation at many places in contracts LaunchpadxIDO.sol and LaunchpadxFactory.sol. Lack of zero address validation has been found at many instances when assigning user supplied address values to state variables directly.

Recommendation:

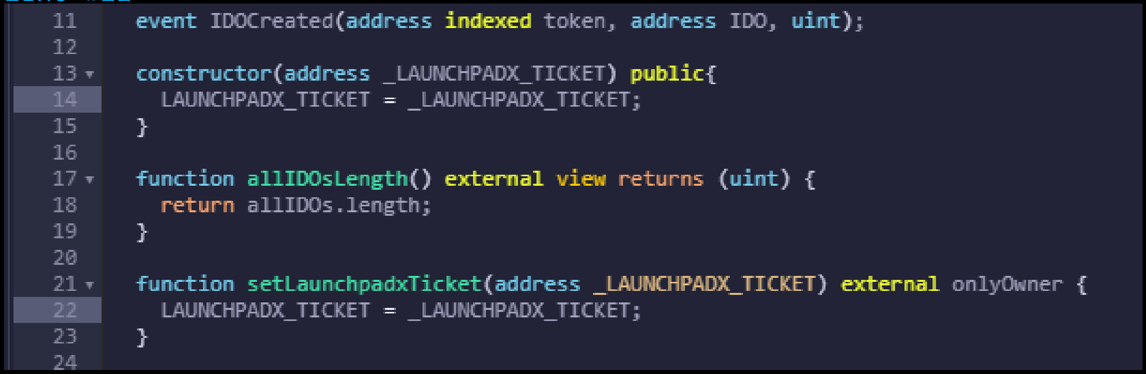
Add proper address validation when every state variable assignment done from user supplied input.

Code Location:

LaunchpadxFactory.sol:

Line #14

Line #22



```
11    event IDOCreated(address indexed token, address IDO, uint);
12
13    constructor(address _LAUNCHPADX_TICKET) public{
14      LAUNCHPADX_TICKET = _LAUNCHPADX_TICKET;
15    }
16
17    function allIDOsLength() external view returns (uint) {
18      return allIDOs.length;
19    }
20
21    function setLaunchpadxTicket(address _LAUNCHPADX_TICKET) external onlyOwner {
22      LAUNCHPADX_TICKET = _LAUNCHPADX_TICKET;
23    }
24
```

LaunchpadxIDO.sol:

Line #80

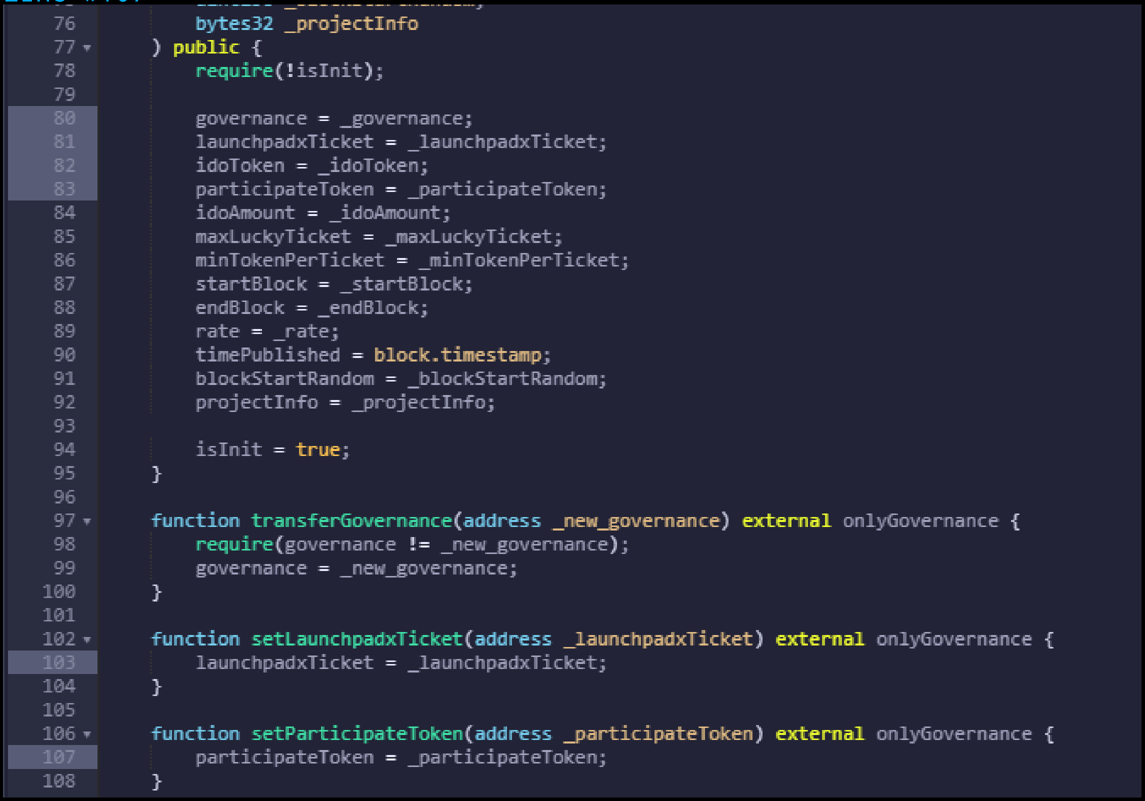
Line #81

Line #82

Line #83

Line #103

Line #107



```
 76             bytes32 _projectInfo
 77         ) public {
 78             require(!isInit);
 79
 80             governance = _governance;
 81             launchpadxTicket = _launchpadxTicket;
 82             idoToken = _idoToken;
 83             participateToken = _participateToken;
 84             idoAmount = _idoAmount;
 85             maxLuckyTicket = _maxLuckyTicket;
 86             minTokenPerTicket = _minTokenPerTicket;
 87             startBlock = _startBlock;
 88             endBlock = _endBlock;
 89             rate = _rate;
 90             timePublished = block.timestamp;
 91             blockStartRandom = _blockStartRandom;
 92             projectInfo = _projectInfo;
 93
 94             isInit = true;
 95         }
 96
 97         function transferGovernance(address _new_governance) external onlyGovernance {
 98             require(governance != _new_governance);
 99             governance = _new_governance;
100         }
101
102         function setLaunchpadxTicket(address _launchpadxTicket) external onlyGovernance {
103             launchpadxTicket = _launchpadxTicket;
104         }
105
106         function setParticipateToken(address _participateToken) external onlyGovernance {
107             participateToken = _participateToken;
108         }
```

3.2 IGNORE RETURN VALUES - INFORMATIONAL

Description:

The return value of an external call is not stored in a local or state variable. In contract LaunchpadxIDO.sol, there are few instances where external methods are being called and return value(bool) are being ignored.

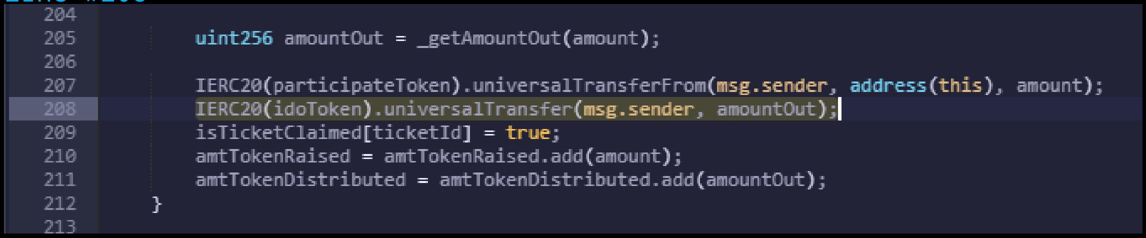
Recommendation:

Add return value check to avoid unexpected crash of the contract. Return value check will help in handling the exceptions better way.

Code Location:

LaunchpadxIDO.sol: Ignoring boolean return type

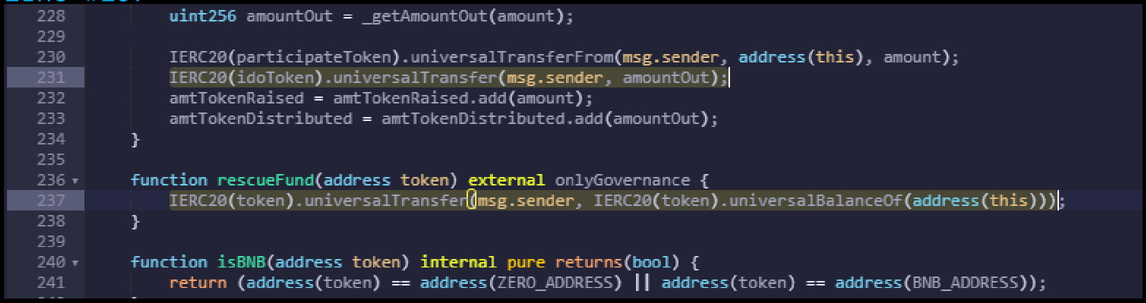
Line #208



```
204
205             uint256 amountOut = _getAmountOut(amount);
206
207             IERC20(participateToken).universalTransferFrom(msg.sender, address(this), amount);
208             IERC20(idoToken).universalTransfer(msg.sender, amountOut);
209             isTicketClaimed[ticketId] = true;
210             amtTokenRaised = amtTokenRaised.add(amount);
211             amtTokenDistributed = amtTokenDistributed.add(amountOut);
212         }
213
```

Line #231

Line #237

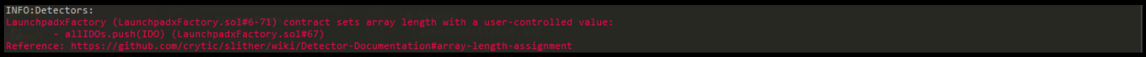


```
228         uint256 amountOut = _getAmountOut(amount);
229
230         IERC20(participateToken).universalTransferFrom(msg.sender, address(this), amount);
231         IERC20(idoToken).universalTransfer(msg.sender, amountOut);
232         amtTokenRaised = amtTokenRaised.add(amount);
233         amtTokenDistributed = amtTokenDistributed.add(amountOut);
234     }
235
236     function rescueFund(address token) external onlyGovernance {
237         IERC20(token).universalTransfer(msg.sender, IERC20(token).universalBalanceOf(address(this)));
238     }
239
240     function isBNB(address token) internal pure returns(bool) {
241         return (address(token) == address(ZERO_ADDRESS) || address(token) == address(BNB_ADDRESS));
```

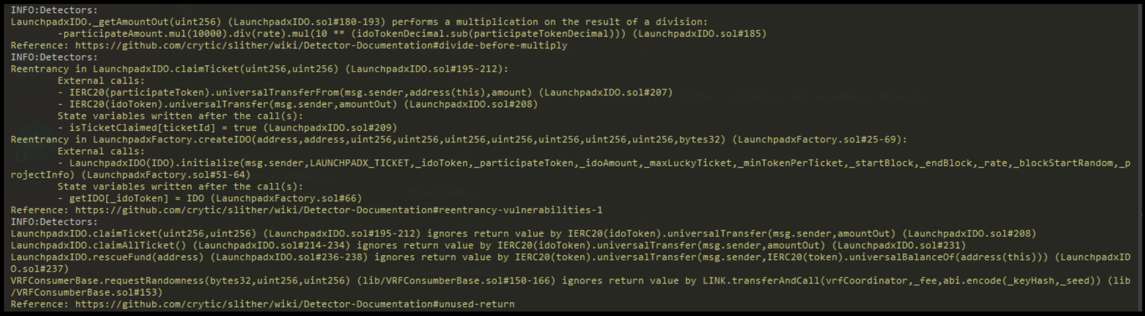
3.3 STATIC ANALYSIS - INFORMATIONAL

Description:

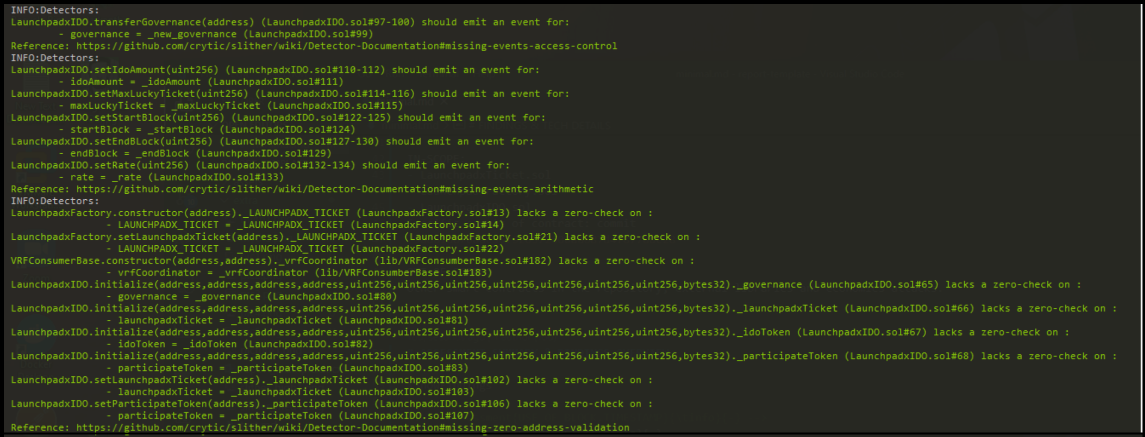
Slither and MythX has been run on all the scoped contracts(LaunchpadxTicket.sol, LaunchpadxFactory.sol and LaunchpadxIDO.sol)



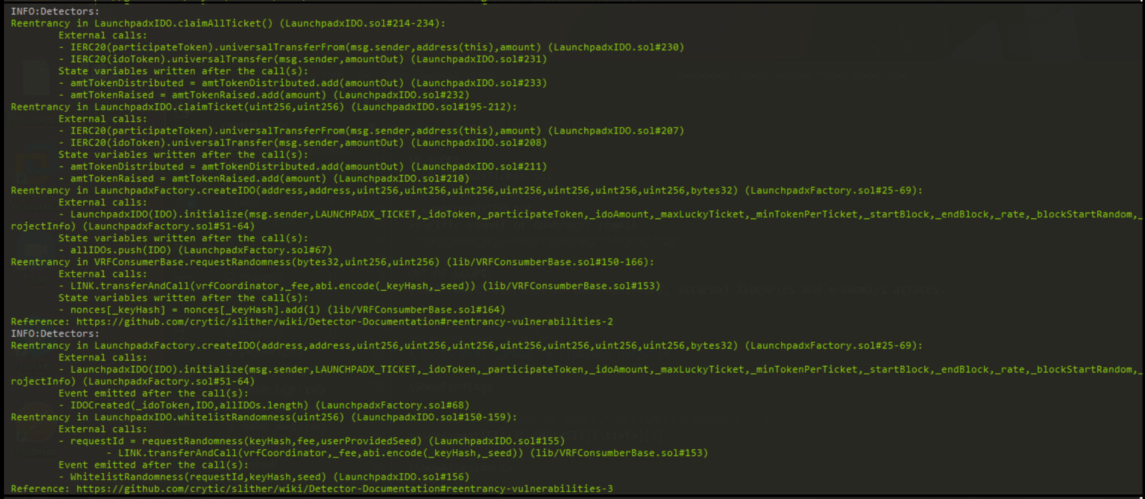
```
INFO:Detectors:
LaunchpadxFactory (LaunchpadxFactory.sol#6-71) contract sets array length with a user-controlled value:
        - allIDOs.push(IDO) (LaunchpadxFactory.sol#67)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#array-length-assignment
```



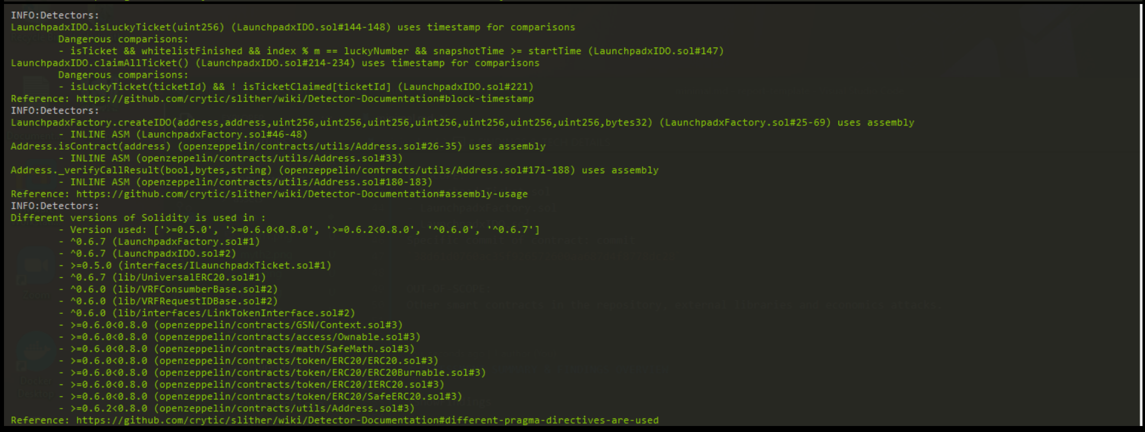
```
INFO:Detectors:
LaunchpadxIDO._getAmountOut(uint256) (LaunchpadxIDO.sol#180-193) performs a multiplication on the result of a division:
        -participateAmount.mul(10000).div(rate).mul(10 ** (idoTokenDecimal.sub(participateTokenDecimal))) (LaunchpadxIDO.sol#185)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#divide-before-multiply
INFO:Detectors:
Reentrancy in LaunchpadxIDO.claimTicket(uint256,uint256) (LaunchpadxIDO.sol#195-212):
        External calls:
        - IERC20(participateToken).universalTransferFrom(msg.sender,address(this),amount) (LaunchpadxIDO.sol#207)
        - IERC20(idoToken).universalTransfer(msg.sender,amountOut) (LaunchpadxIDO.sol#208)
        State variables written after the call(s):
        - isTicketClaimed[ticketId] = true (LaunchpadxIDO.sol#209)
Reentrancy in LaunchpadxFactory.createIDO(address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32) (LaunchpadxFactory.sol#25-69):
        External calls:
        - LaunchpadxIDO(IDO).initialize(msg.sender,LAUNCHPADX_TICKET,_idoToken,_participateToken,_idoAmount,_maxLuckyTicket,_minTokenPerTicket,_startBlock,_endBlock,_rate,_blockStartRandom,_p
rojectInfo) (LaunchpadxFactory.sol#51-64)
        State variables written after the call(s):
        - getIDO[_idoToken] = IDO (LaunchpadxFactory.sol#66)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-vulnerabilities-1
INFO:Detectors:
LaunchpadxIDO.claimTicket(uint256,uint256) (LaunchpadxIDO.sol#195-212) ignores return value by IERC20(idoToken).universalTransfer(msg.sender,amountOut) (LaunchpadxIDO.sol#208)
LaunchpadxIDO.claimAllTicket() (LaunchpadxIDO.sol#214-234) ignores return value by IERC20(idoToken).universalTransfer(msg.sender,amountOut) (LaunchpadxIDO.sol#231)
LaunchpadxIDO.rescueFund(address) (LaunchpadxIDO.sol#236-238) ignores return value by IERC20(token).universalTransfer(msg.sender,IERC20(token).universalBalanceOf(address(this))) (LaunchpadxID
O.sol#237)
VRFConsumerBase.requestRandomness(bytes32,uint256,uint256) (lib/VRFConsumberBase.sol#150-166) ignores return value by LINK.transferAndCall(vrfCoordinator,_fee,abi.encode(_keyHash,_seed)) (lib
/VRFConsumberBase.sol#153)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#unused-return
```



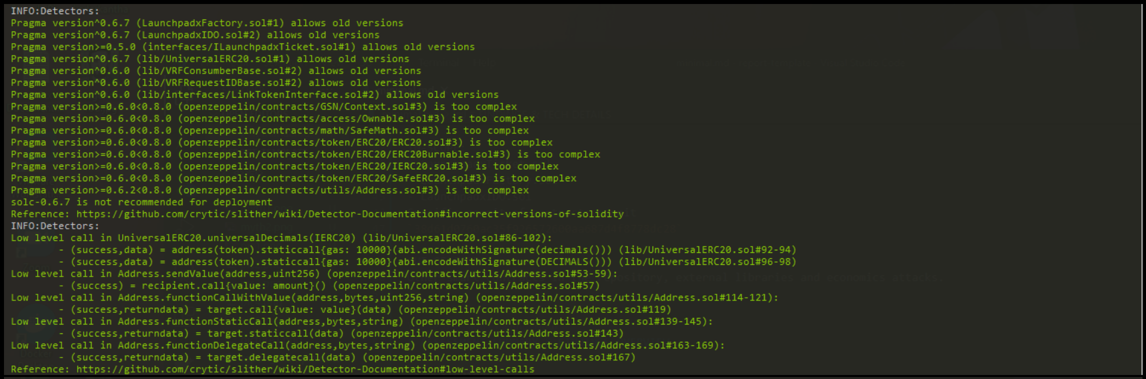
```
INFO:Detectors:
LaunchpadxIDO.transferGovernance(address) (LaunchpadxIDO.sol#97-100) should emit an event for:
        - governance = _new_governance (LaunchpadxIDO.sol#99)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#missing-events-access-control
INFO:Detectors:
LaunchpadxIDO.setIdoAmount(uint256) (LaunchpadxIDO.sol#110-112) should emit an event for:
        - idoAmount = _idoAmount (LaunchpadxIDO.sol#111)
LaunchpadxIDO.setMaxLuckyTicket(uint256) (LaunchpadxIDO.sol#114-116) should emit an event for:
        - maxLuckyTicket = _maxLuckyTicket (LaunchpadxIDO.sol#115)
LaunchpadxIDO.setStartBlock(uint256) (LaunchpadxIDO.sol#122-125) should emit an event for:
        - startBlock = _startBlock (LaunchpadxIDO.sol#124)
LaunchpadxIDO.setEndBLock(uint256) (LaunchpadxIDO.sol#127-130) should emit an event for:
        - endBlock = _endBlock (LaunchpadxIDO.sol#129)
LaunchpadxIDO.setRate(uint256) (LaunchpadxIDO.sol#132-134) should emit an event for:
        - rate = _rate (LaunchpadxIDO.sol#133)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#missing-events-arithmetic
INFO:Detectors:
LaunchpadxFactory.constructor(address)._LAUNCHPADX_TICKET (LaunchpadxFactory.sol#13) lacks a zero-check on :
                - LAUNCHPADX_TICKET = _LAUNCHPADX_TICKET (LaunchpadxFactory.sol#14)
LaunchpadxFactory.setLaunchpadxTicket(address)._LAUNCHPADX_TICKET (LaunchpadxFactory.sol#21) lacks a zero-check on :
                - LAUNCHPADX_TICKET = _LAUNCHPADX_TICKET (LaunchpadxFactory.sol#22)
VRFConsumerBase.constructor(address,address)._vrfCoordinator (lib/VRFConsumberBase.sol#182) lacks a zero-check on :
                - vrfCoordinator = _vrfCoordinator (lib/VRFConsumberBase.sol#183)
LaunchpadxIDO.initialize(address,address,address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._governance (LaunchpadxIDO.sol#65) lacks a zero-check on :
                - governance = _governance (LaunchpadxIDO.sol#80)
LaunchpadxIDO.initialize(address,address,address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._launchpadxTicket (LaunchpadxIDO.sol#66) lacks a zero-check on :
                - launchpadxTicket = _launchpadxTicket (LaunchpadxIDO.sol#81)
LaunchpadxIDO.initialize(address,address,address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._idoToken (LaunchpadxIDO.sol#67) lacks a zero-check on :
                - idoToken = _idoToken (LaunchpadxIDO.sol#82)
LaunchpadxIDO.initialize(address,address,address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._participateToken (LaunchpadxIDO.sol#68) lacks a zero-check on :
                - participateToken = _participateToken (LaunchpadxIDO.sol#83)
LaunchpadxIDO.setLaunchpadxTicket(address)._launchpadxTicket (LaunchpadxIDO.sol#102) lacks a zero-check on :
                - launchpadxTicket = _launchpadxTicket (LaunchpadxIDO.sol#103)
LaunchpadxIDO.setParticipateToken(address)._participateToken (LaunchpadxIDO.sol#106) lacks a zero-check on :
                - participateToken = _participateToken (LaunchpadxIDO.sol#107)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#missing-zero-address-validation
```



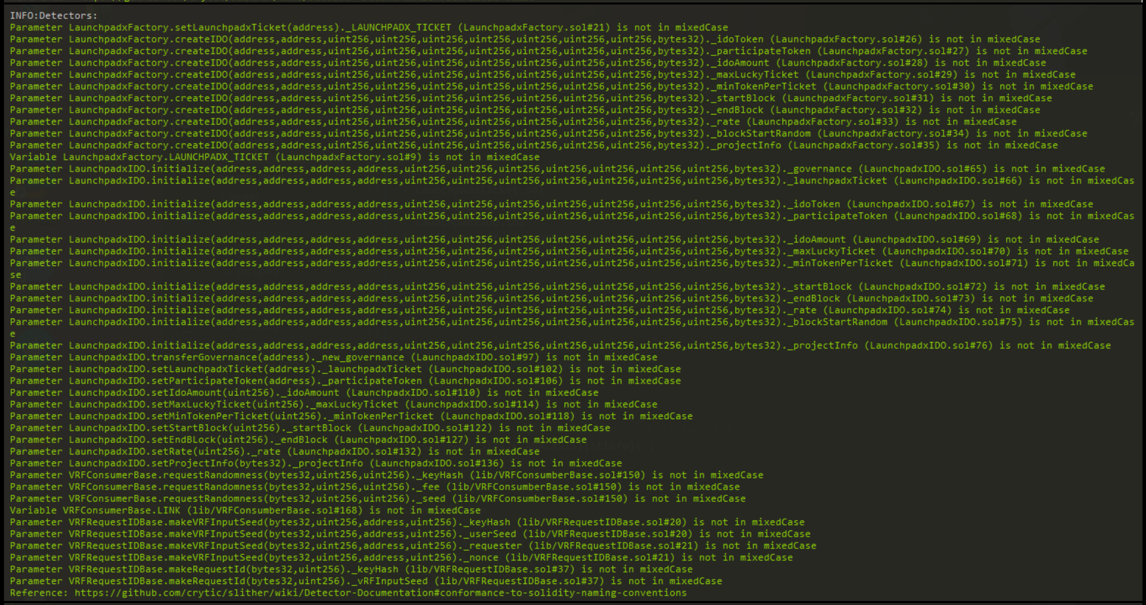
```
INFO:Detectors:
Reentrancy in LaunchpadxIDO.claimAllTicket() (LaunchpadxIDO.sol#214-234):
        External calls:
        - IERC20(participateToken).universalTransferFrom(msg.sender,address(this),amount) (LaunchpadxIDO.sol#230)
        - IERC20(idoToken).universalTransfer(msg.sender,amountOut) (LaunchpadxIDO.sol#231)
        State variables written after the call(s):
        - amtTokenDistributed = amtTokenDistributed.add(amountOut) (LaunchpadxIDO.sol#233)
        - amtTokenRaised = amtTokenRaised.add(amount) (LaunchpadxIDO.sol#232)
Reentrancy in LaunchpadxIDO.claimTicket(uint256,uint256) (LaunchpadxIDO.sol#195-212):
        External calls:
        - IERC20(participateToken).universalTransferFrom(msg.sender,address(this),amount) (LaunchpadxIDO.sol#207)
        - IERC20(idoToken).universalTransfer(msg.sender,amountOut) (LaunchpadxIDO.sol#208)
        State variables written after the call(s):
        - amtTokenDistributed = amtTokenDistributed.add(amountOut) (LaunchpadxIDO.sol#211)
        - amtTokenRaised = amtTokenRaised.add(amount) (LaunchpadxIDO.sol#210)
Reentrancy in LaunchpadxFactory.createIDO(address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32) (LaunchpadxFactory.sol#25-69):
        External calls:
        - LaunchpadxIDO(IDO).initialize(msg.sender,LAUNCHPADX_TICKET,_idoToken,_participateToken,_idoAmount,_maxLuckyTicket,_minTokenPerTicket,_startBlock,_endBlock,_rate,_blockStartRandom,_
rojectInfo) (LaunchpadxFactory.sol#51-64)
        State variables written after the call(s):
        - allIDOs.push(IDO) (LaunchpadxFactory.sol#67)
Reentrancy in VRFConsumerBase.requestRandomness(bytes32,uint256,uint256) (lib/VRFConsumberBase.sol#150-166):
        External calls:
        - LINK.transferAndCall(vrfCoordinator,_fee,abi.encode(_keyHash,_seed)) (lib/VRFConsumberBase.sol#153)
        State variables written after the call(s):
        - nonces[_keyHash] = nonces[_keyHash].add(1) (lib/VRFConsumberBase.sol#164)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-vulnerabilities-2
INFO:Detectors:
Reentrancy in LaunchpadxFactory.createIDO(address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32) (LaunchpadxFactory.sol#25-69):
        External calls:
        - LaunchpadxIDO(IDO).initialize(msg.sender,LAUNCHPADX_TICKET,_idoToken,_participateToken,_idoAmount,_maxLuckyTicket,_minTokenPerTicket,_startBlock,_endBlock,_rate,_blockStartRandom,_
rojectInfo) (LaunchpadxFactory.sol#51-64)
        Event emitted after the call(s):
        - IDOCreated(_idoToken,IDO,allIDOs.length) (LaunchpadxFactory.sol#68)
Reentrancy in LaunchpadxIDO.whitelistRandomness(uint256) (LaunchpadxIDO.sol#150-159):
        External calls:
        - requestId = requestRandomness(keyHash,fee,userProvidedSeed) (LaunchpadxIDO.sol#155)
                - LINK.transferAndCall(vrfCoordinator,_fee,abi.encode(_keyHash,_seed)) (lib/VRFConsumberBase.sol#153)
        Event emitted after the call(s):
        - WhitelistRandomness(requestId,keyHash,seed) (LaunchpadxIDO.sol#156)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-vulnerabilities-3
```



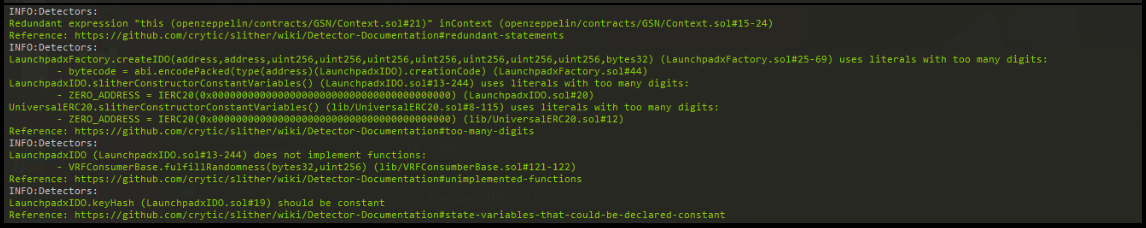
```
INFO:Detectors:
LaunchpadxIDO.isLuckyTicket(uint256) (LaunchpadxIDO.sol#144-148) uses timestamp for comparisons
        Dangerous comparisons:
        - isTicket && whitelistFinished && index % m == luckyNumber && snapshotTime >= startTime (LaunchpadxIDO.sol#147)
LaunchpadxIDO.claimAllTicket() (LaunchpadxIDO.sol#214-234) uses timestamp for comparisons
        Dangerous comparisons:
        - isLuckyTicket(ticketId) && ! isTicketClaimed[ticketId] (LaunchpadxIDO.sol#221)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#block-timestamp
INFO:Detectors:
LaunchpadxFactory.createIDO(address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32) (LaunchpadxFactory.sol#25-69) uses assembly
        - INLINE ASM (LaunchpadxFactory.sol#46-48)
Address.isContract(address) (openzeppelin/contracts/utils/Address.sol#26-35) uses assembly
        - INLINE ASM (openzeppelin/contracts/utils/Address.sol#33)
Address._verifyCallResult(bool,bytes,string) (openzeppelin/contracts/utils/Address.sol#171-188) uses assembly
        - INLINE ASM (openzeppelin/contracts/utils/Address.sol#180-183)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#assembly-usage
INFO:Detectors:
Different versions of Solidity is used in :
        - Version used: ['>=0.5.0', '>=0.6.0<0.8.0', '>=0.6.2<0.8.0', '^0.6.0', '^0.6.7']
        - ^0.6.7 (LaunchpadxFactory.sol#1)
        - ^0.6.7 (LaunchpadxIDO.sol#2)
        - >=0.5.0 (interfaces/ILaunchpadxTicket.sol#1)
        - ^0.6.7 (lib/UniversalERC20.sol#1)
        - ^0.6.0 (lib/VRFConsumberBase.sol#2)
        - ^0.6.0 (lib/VRFRequestIDBase.sol#2)
        - ^0.6.0 (lib/interfaces/LinkTokenInterface.sol#2)
        - >=0.6.0<0.8.0 (openzeppelin/contracts/GSN/Context.sol#3)
        - >=0.6.0<0.8.0 (openzeppelin/contracts/access/Ownable.sol#3)
        - >=0.6.0<0.8.0 (openzeppelin/contracts/math/SafeMath.sol#3)
        - >=0.6.0<0.8.0 (openzeppelin/contracts/token/ERC20/ERC20.sol#3)
        - >=0.6.0<0.8.0 (openzeppelin/contracts/token/ERC20/ERC20Burnable.sol#3)
        - >=0.6.0<0.8.0 (openzeppelin/contracts/token/ERC20/IERC20.sol#3)
        - >=0.6.0<0.8.0 (openzeppelin/contracts/token/ERC20/SafeERC20.sol#3)
        - >=0.6.2<0.8.0 (openzeppelin/contracts/utils/Address.sol#3)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#different-pragma-directives-are-used
```



```
INFO:Detectors:
Pragma version^0.6.7 (LaunchpadxFactory.sol#1) allows old versions
Pragma version^0.6.7 (LaunchpadxIDO.sol#2) allows old versions
Pragma version>=0.5.0 (interfaces/ILaunchpadxTicket.sol#1) allows old versions
Pragma version^0.6.7 (lib/UniversalERC20.sol#1) allows old versions
Pragma version^0.6.0 (lib/VRFConsumberBase.sol#2) allows old versions
Pragma version^0.6.0 (lib/VRFRequestIDBase.sol#2) allows old versions
Pragma version^0.6.0 (lib/interfaces/LinkTokenInterface.sol#2) allows old versions
Pragma version>=0.6.0<0.8.0 (openzeppelin/contracts/GSN/Context.sol#3) is too complex
Pragma version>=0.6.0<0.8.0 (openzeppelin/contracts/access/Ownable.sol#3) is too complex
Pragma version>=0.6.0<0.8.0 (openzeppelin/contracts/math/SafeMath.sol#3) is too complex
Pragma version>=0.6.0<0.8.0 (openzeppelin/contracts/token/ERC20/ERC20.sol#3) is too complex
Pragma version>=0.6.0<0.8.0 (openzeppelin/contracts/token/ERC20/ERC20Burnable.sol#3) is too complex
Pragma version>=0.6.0<0.8.0 (openzeppelin/contracts/token/ERC20/IERC20.sol#3) is too complex
Pragma version>=0.6.0<0.8.0 (openzeppelin/contracts/token/ERC20/SafeERC20.sol#3) is too complex
Pragma version>=0.6.2<0.8.0 (openzeppelin/contracts/utils/Address.sol#3) is too complex
solc-0.6.7 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-versions-of-solidity
INFO:Detectors:
Low level call in UniversalERC20.universalDecimals(IERC20) (lib/UniversalERC20.sol#86-102):
        - (success,data) = address(token).staticcall{gas: 10000}(abi.encodeWithSignature(decimals())) (lib/UniversalERC20.sol#92-94)
        - (success,data) = address(token).staticcall{gas: 10000}(abi.encodeWithSignature(DECIMALS())) (lib/UniversalERC20.sol#96-98)
Low level call in Address.sendValue(address,uint256) (openzeppelin/contracts/utils/Address.sol#53-59):
        - (success) = recipient.call{value: amount}() (openzeppelin/contracts/utils/Address.sol#57)
Low level call in Address.functionCallWithValue(address,bytes,uint256,string) (openzeppelin/contracts/utils/Address.sol#114-121):
        - (success,returndata) = target.call{value: value}(data) (openzeppelin/contracts/utils/Address.sol#119)
Low level call in Address.functionStaticCall(address,bytes,string) (openzeppelin/contracts/utils/Address.sol#139-145):
        - (success,returndata) = target.staticcall(data) (openzeppelin/contracts/utils/Address.sol#143)
Low level call in Address.functionDelegateCall(address,bytes,string) (openzeppelin/contracts/utils/Address.sol#163-169):
        - (success,returndata) = target.delegatecall(data) (openzeppelin/contracts/utils/Address.sol#167)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#low-level-calls
```



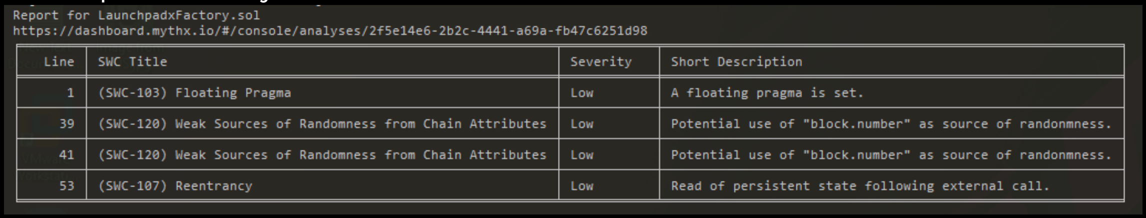
```
INFO:Detectors:
Parameter LaunchpadxFactory.setLaunchpadxTicket(address)._LAUNCHPADX_TICKET (LaunchpadxFactory.sol#21) is not in mixedCase
Parameter LaunchpadxFactory.createIDO(address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._idoToken (LaunchpadxFactory.sol#26) is not in mixedCase
Parameter LaunchpadxFactory.createIDO(address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._participateToken (LaunchpadxFactory.sol#27) is not in mixedCase
Parameter LaunchpadxFactory.createIDO(address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._idoAmount (LaunchpadxFactory.sol#28) is not in mixedCase
Parameter LaunchpadxFactory.createIDO(address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._maxLuckyTicket (LaunchpadxFactory.sol#29) is not in mixedCase
Parameter LaunchpadxFactory.createIDO(address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._minTokenPerTicket (LaunchpadxFactory.sol#30) is not in mixedCase
Parameter LaunchpadxFactory.createIDO(address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._startBlock (LaunchpadxFactory.sol#31) is not in mixedCase
Parameter LaunchpadxFactory.createIDO(address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._endBlock (LaunchpadxFactory.sol#32) is not in mixedCase
Parameter LaunchpadxFactory.createIDO(address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._rate (LaunchpadxFactory.sol#33) is not in mixedCase
Parameter LaunchpadxFactory.createIDO(address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._blockStartRandom (LaunchpadxFactory.sol#34) is not in mixedCase
Parameter LaunchpadxFactory.createIDO(address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._projectInfo (LaunchpadxFactory.sol#35) is not in mixedCase
Variable LaunchpadxFactory.LAUNCHPADX_TICKET (LaunchpadxFactory.sol#9) is not in mixedCase
Parameter LaunchpadxIDO.initialize(address,address,address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._governance (LaunchpadxIDO.sol#65) is not in mixedCase
Parameter LaunchpadxIDO.initialize(address,address,address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._launchpadxTicket (LaunchpadxIDO.sol#66) is not in mixedCas
e
Parameter LaunchpadxIDO.initialize(address,address,address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._idoToken (LaunchpadxIDO.sol#67) is not in mixedCase
Parameter LaunchpadxIDO.initialize(address,address,address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._participateToken (LaunchpadxIDO.sol#68) is not in mixedCas
e
Parameter LaunchpadxIDO.initialize(address,address,address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._idoAmount (LaunchpadxIDO.sol#69) is not in mixedCase
Parameter LaunchpadxIDO.initialize(address,address,address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._maxLuckyTicket (LaunchpadxIDO.sol#70) is not in mixedCase
Parameter LaunchpadxIDO.initialize(address,address,address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._minTokenPerTicket (LaunchpadxIDO.sol#71) is not in mixedCa
se
Parameter LaunchpadxIDO.initialize(address,address,address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._startBlock (LaunchpadxIDO.sol#72) is not in mixedCase
Parameter LaunchpadxIDO.initialize(address,address,address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._endBlock (LaunchpadxIDO.sol#73) is not in mixedCase
Parameter LaunchpadxIDO.initialize(address,address,address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._rate (LaunchpadxIDO.sol#74) is not in mixedCase
Parameter LaunchpadxIDO.initialize(address,address,address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._blockStartRandom (LaunchpadxIDO.sol#75) is not in mixedCas
e
Parameter LaunchpadxIDO.initialize(address,address,address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32)._projectInfo (LaunchpadxIDO.sol#76) is not in mixedCase
Parameter LaunchpadxIDO.transferGovernance(address)._new_governance (LaunchpadxIDO.sol#97) is not in mixedCase
Parameter LaunchpadxIDO.setLaunchpadxTicket(address)._launchpadxTicket (LaunchpadxIDO.sol#102) is not in mixedCase
Parameter LaunchpadxIDO.setParticipateToken(address)._participateToken (LaunchpadxIDO.sol#106) is not in mixedCase
Parameter LaunchpadxIDO.setIdoAmount(uint256)._idoAmount (LaunchpadxIDO.sol#110) is not in mixedCase
Parameter LaunchpadxIDO.setMaxLuckyTicket(uint256)._maxLuckyTicket (LaunchpadxIDO.sol#114) is not in mixedCase
Parameter LaunchpadxIDO.setMinTokenPerTicket(uint256)._minTokenPerTicket (LaunchpadxIDO.sol#118) is not in mixedCase
Parameter LaunchpadxIDO.setStartBlock(uint256)._startBlock (LaunchpadxIDO.sol#122) is not in mixedCase
Parameter LaunchpadxIDO.setEndBLock(uint256)._endBlock (LaunchpadxIDO.sol#127) is not in mixedCase
Parameter LaunchpadxIDO.setRate(uint256)._rate (LaunchpadxIDO.sol#132) is not in mixedCase
Parameter LaunchpadxIDO.setProjectInfo(bytes32)._projectInfo (LaunchpadxIDO.sol#136) is not in mixedCase
Parameter VRFConsumerBase.requestRandomness(bytes32,uint256,uint256)._keyHash (lib/VRFConsumberBase.sol#150) is not in mixedCase
Parameter VRFConsumerBase.requestRandomness(bytes32,uint256,uint256)._fee (lib/VRFConsumberBase.sol#150) is not in mixedCase
Parameter VRFConsumerBase.requestRandomness(bytes32,uint256,uint256)._seed (lib/VRFConsumberBase.sol#150) is not in mixedCase
Variable VRFConsumerBase.LINK (lib/VRFConsumberBase.sol#168) is not in mixedCase
Parameter VRFRequestIDBase.makeVRFInputSeed(bytes32,uint256,address,uint256)._keyHash (lib/VRFRequestIDBase.sol#20) is not in mixedCase
Parameter VRFRequestIDBase.makeVRFInputSeed(bytes32,uint256,address,uint256)._userSeed (lib/VRFRequestIDBase.sol#20) is not in mixedCase
Parameter VRFRequestIDBase.makeVRFInputSeed(bytes32,uint256,address,uint256)._requester (lib/VRFRequestIDBase.sol#21) is not in mixedCase
Parameter VRFRequestIDBase.makeVRFInputSeed(bytes32,uint256,address,uint256)._nonce (lib/VRFRequestIDBase.sol#21) is not in mixedCase
Parameter VRFRequestIDBase.makeRequestId(bytes32,uint256)._keyHash (lib/VRFRequestIDBase.sol#37) is not in mixedCase
Parameter VRFRequestIDBase.makeRequestId(bytes32,uint256)._vRFInputSeed (lib/VRFRequestIDBase.sol#37) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-solidity-naming-conventions
```



```
INFO:Detectors:
Redundant expression "this (openzeppelin/contracts/GSN/Context.sol#21)" inContext (openzeppelin/contracts/GSN/Context.sol#15-24)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#redundant-statements
INFO:Detectors:
LaunchpadxFactory.createIDO(address,address,uint256,uint256,uint256,uint256,uint256,uint256,uint256,bytes32) (LaunchpadxFactory.sol#25-69) uses literals with too many digits:
        - bytecode = abi.encodePacked(type(address)(LaunchpadxIDO).creationCode) (LaunchpadxFactory.sol#44)
LaunchpadxIDO.slitherConstructorConstantVariables() (LaunchpadxIDO.sol#13-244) uses literals with too many digits:
        - ZERO_ADDRESS = IERC20(0x0000000000000000000000000000000000000000) (LaunchpadxIDO.sol#20)
UniversalERC20.slitherConstructorConstantVariables() (lib/UniversalERC20.sol#8-115) uses literals with too many digits:
        - ZERO_ADDRESS = IERC20(0x0000000000000000000000000000000000000000) (lib/UniversalERC20.sol#12)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#too-many-digits
INFO:Detectors:
LaunchpadxIDO (LaunchpadxIDO.sol#13-244) does not implement functions:
        - VRFConsumerBase.fulfillRandomness(bytes32,uint256) (lib/VRFConsumberBase.sol#121-122)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#unimplemented-functions
INFO:Detectors:
LaunchpadxIDO.keyHash (LaunchpadxIDO.sol#19) should be constant
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-constant
```

MythX:

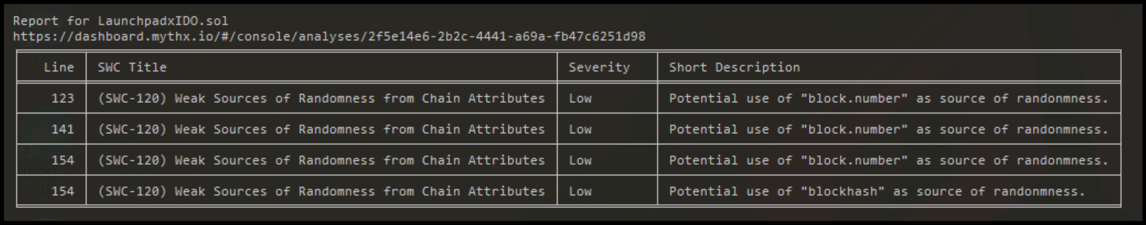
LaunchpadxFactory.sol



Report for LaunchpadxFactory.sol
https://dashboard.mythx.io/#/console/analyses/2f5e14e6-2b2c-4441-a69a-fb47c6251d98

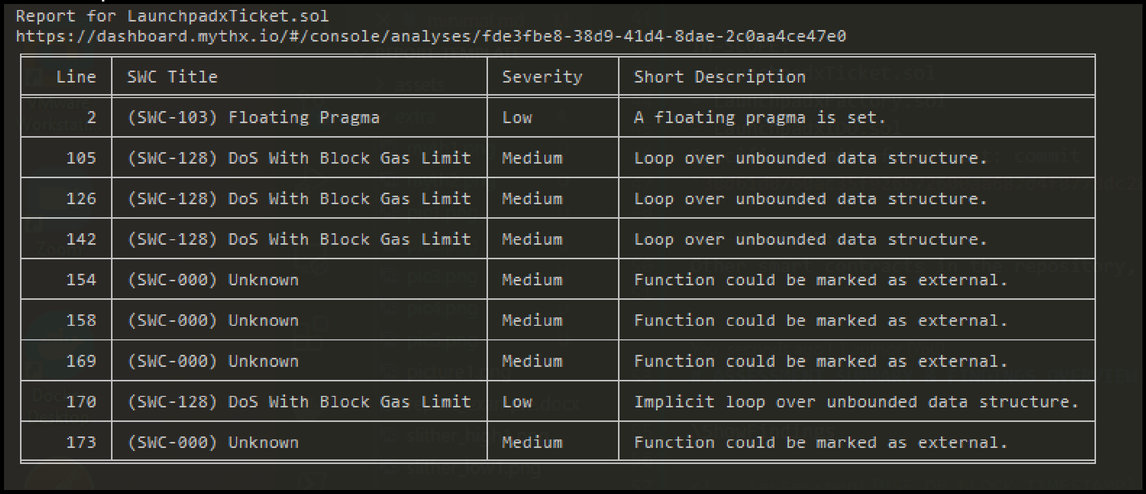
Line	SWC Title	Severity	Short Description
1	(SWC-103) Floating Pragma	Low	A floating pragma is set.
39	(SWC-120) Weak Sources of Randomness from Chain Attributes	Low	Potential use of "block.number" as source of randonmness.
41	(SWC-120) Weak Sources of Randomness from Chain Attributes	Low	Potential use of "block.number" as source of randonmness.
53	(SWC-107) Reentrancy	Low	Read of persistent state following external call.

LauncpadxIDO.sol



```
Report for LaunchpadxIDO.sol
https://dashboard.mythx.io/#/console/analyses/2f5e14e6-2b2c-4441-a69a-fb47c6251d98
╒════════╤════════════════════════════════════════════════════════╤════════════╤══════════════════════════════════════════════════════════╕
│   Line │ SWC Title                                              │ Severity   │ Short Description                                        │
╞════════╪════════════════════════════════════════════════════════╪════════════╪══════════════════════════════════════════════════════════╡
│    123 │ (SWC-120) Weak Sources of Randomness from Chain Attributes │ Low    │ Potential use of "block.number" as source of randonmness. │
├────────┼────────────────────────────────────────────────────────┼────────────┼──────────────────────────────────────────────────────────┤
│    141 │ (SWC-120) Weak Sources of Randomness from Chain Attributes │ Low    │ Potential use of "block.number" as source of randonmness. │
├────────┼────────────────────────────────────────────────────────┼────────────┼──────────────────────────────────────────────────────────┤
│    154 │ (SWC-120) Weak Sources of Randomness from Chain Attributes │ Low    │ Potential use of "block.number" as source of randonmness. │
├────────┼────────────────────────────────────────────────────────┼────────────┼──────────────────────────────────────────────────────────┤
│    154 │ (SWC-120) Weak Sources of Randomness from Chain Attributes │ Low    │ Potential use of "blockhash" as source of randonmness.    │
╘════════╧════════════════════════════════════════════════════════╧════════════╧══════════════════════════════════════════════════════════╛
```

LaunchpadxTicket.sol



```
Report for LaunchpadxTicket.sol
https://dashboard.mythx.io/#/console/analyses/fde3fbe8-38d9-41d4-8dae-2c0aa4ce47e0
╒════════╤════════════════════════════════╤════════════╤════════════════════════════════════════════╕
│   Line │ SWC Title                      │ Severity   │ Short Description                          │
╞════════╪════════════════════════════════╪════════════╪════════════════════════════════════════════╡
│      2 │ (SWC-103) Floating Pragma      │ Low        │ A floating pragma is set.                  │
├────────┼────────────────────────────────┼────────────┼────────────────────────────────────────────┤
│    105 │ (SWC-128) DoS With Block Gas Limit │ Medium │ Loop over unbounded data structure.        │
├────────┼────────────────────────────────┼────────────┼────────────────────────────────────────────┤
│    126 │ (SWC-128) DoS With Block Gas Limit │ Medium │ Loop over unbounded data structure.        │
├────────┼────────────────────────────────┼────────────┼────────────────────────────────────────────┤
│    142 │ (SWC-128) DoS With Block Gas Limit │ Medium │ Loop over unbounded data structure.        │
├────────┼────────────────────────────────┼────────────┼────────────────────────────────────────────┤
│    154 │ (SWC-000) Unknown              │ Medium     │ Function could be marked as external.      │
├────────┼────────────────────────────────┼────────────┼────────────────────────────────────────────┤
│    158 │ (SWC-000) Unknown              │ Medium     │ Function could be marked as external.      │
├────────┼────────────────────────────────┼────────────┼────────────────────────────────────────────┤
│    169 │ (SWC-000) Unknown              │ Medium     │ Function could be marked as external.      │
├────────┼────────────────────────────────┼────────────┼────────────────────────────────────────────┤
│    170 │ (SWC-128) DoS With Block Gas Limit │ Low    │ Implicit loop over unbounded data structure. │
├────────┼────────────────────────────────┼────────────┼────────────────────────────────────────────┤
│    173 │ (SWC-000) Unknown              │ Medium     │ Function could be marked as external.      │
╘════════╧════════════════════════════════╧════════════╧════════════════════════════════════════════╛
```

THANK YOU FOR CHOOSING

// HALBORN