Smart Contract Vulnerability Audit

DCM

Dec 15, 2021

Smart Contract – Audit Overview

Project Summary

Project Name	DCM
Platform	Binance Smart Chain
Language	Solidity
Commits	0x1a2bf408934e265c5f28ebd7dd2d3f61f6e61347

Audit Summary

Delivery Date	December 16, 2021
Method of Audit	Human and AI
Consultants Engaged	One
Timeline	December 15, 2021 – December 16, 2021

Vulnerability Summary

Vulnerability Level	Total	Resolved
Critical	0	✓
Major	0	✓
Medium	0	✓
Minor	0	✓

Smart Contract - Contract Overview

All information is recorded as of 12/15/2021.

Contract Name	DCM.sol
Contract Ticker	DCM
Contract Address	0x1A2Bf408934E265c5f28EBD7DD2D3f61f6e61347
Contract Creator	0xCe400C96a110f940571CA82e568fbddbD9E7Ee7B
Decimals	9
Total Supply	2,000,000,000
Token Holders	1
Token Transfers	1
Complier Version	v0.8.7+commit.e28d00a7
Source Code	Solidity
Optimization Enabled	No with 200 runs
Other Settings	default evmVersion, MIT license

Smart Contract - Vulnerabilities

Vulnerability Tested	Human Review	Ai Review	Line(s) Affected	Results
Function Default Visibility	✓	✓		✓
Integer Overflow and Underflow	✓	✓		✓
Outdated Compiler Version	✓	✓		✓
Unchecked Call Return Value	✓	✓		✓
Unprotected Ether Withdrawal	✓	✓		✓
Unprotected SELFDESTRUCT Instruction	✓	✓		✓
Unencrypted Private Data On-Chain	✓	✓		✓

Smart Contract - Vulnerabilities

Vulnerability Tested	Human Review	Ai Review	Line(s) Affected	Results
Reentrancy	✓	✓		✓
Uninitialized Storage Pointer	✓	✓		✓
Assert Violation	✓	✓		✓
Use of Deprecated Solidity Functions	✓	✓		✓
Delegatecall to Untrusted Callee	✓	✓		✓
DoS with Failed Call	✓	✓		✓
Code With No Effects	✓	✓		✓

Smart Contract - Vulnerabilities

Vulnerability Tested	Human Review	Ai Review	Line(s) Affected	Results
Missing Protection against Signature Replay Attacks	✓	✓		✓
Lack of Proper Signature Verification	✓	✓		✓
Requirement Violation	✓	✓		✓
Write to Arbitrary Storage Location	✓	✓		✓
Incorrect Inheritance Order	✓	✓		✓
Insufficient Gas Griefing	✓	✓		✓
Arbitrary Jump with Function Type Variable	✓	✓		✓

Smart Contract - Vulnerabilities

Vulnerability Tested	Human Review	Ai Review	Line(s) Affected	Results
DoS With Block Gas Limit	✓	✓		✓
Typographical Error	✓	✓		✓
Right-To-Left-Override control character	✓	✓		✓
Presence of unused variables	✓	✓		✓
Unexpected Ether balance	✓	✓		✓
Hash Collisions With Multiple Variable Length Arguments	✓	✓		✓
Message call with hardcoded gas amount	✓	✓		✓

Smart Contract - Vulnerabilities

Vulnerability Tested	Human Review	Ai Review	Line(s) Affected	Results
Transaction Order Dependence	✓	✓		✓
Block values as a proxy for time	✓	✓		✓
Signature Malleability	✓	✓		✓
Incorrect Constructor Name	✓	✓		✓
Shadowing State Variables	✓	✓		✓
Weak Sources of Randomness from Chain Attributes	✓	✓		✓

Smart Contract - Code Analysis

We did not identify any minor nor significant vulnerabilities within the contract code.

Smart Contract - Contract Functions

+ Context
  - [Int] _msgSender
  - [Int] _msgData

+ [Int] IERC20
  - [Ext] totalSupply
  - [Ext] balanceOf
  - [Ext] transfer #
  - [Ext] allowance
  - [Ext] approve #
  - [Ext] transferFrom #

+ [Lib] SafeMath
  - [Int] add
  - [Int] sub
  - [Int] sub
  - [Int] mul
  - [Int] div
  - [Int] div
  - [Int] mod
  - [Int] mod

+ [Lib] Address
  - [Int] isContract
  - [Int] sendValue #
  - [Int] functionCall #
  - [Int] functionCall #
  - [Int] functionCallWithValue #
  - [Int] functionCallWithValue #
  - [Prv] _functionCallWithValue #

+ Ownable (Context)
  - [Pub] <Constructor> #
  - [Pub] owner
  - [Pub] waiveOwnership #
    - modifiers: onlyOwner
  - [Pub] transferOwnership #
    - modifiers: onlyOwner
  - [Pub] getUnlockTime
  - [Pub] getTime
  - [Pub] lock #
    - modifiers: onlyOwner
  - [Pub] unlock #

Smart Contract - Contract Functions

+ [Int] IUniswapV2Factory
  - [Ext] feeTo
  - [Ext] feeToSetter
  - [Ext] getPair
  - [Ext] allPairs
  - [Ext] allPairsLength
  - [Ext] createPair #
  - [Ext] setFeeTo #
  - [Ext] setFeeToSetter #

+ [Int] IUniswapV2Pair
  - [Ext] name
  - [Ext] symbol
  - [Ext] decimals
  - [Ext] totalSupply
  - [Ext] balanceOf
  - [Ext] allowance
  - [Ext] approve #
  - [Ext] transfer #
  - [Ext] transferFrom #
  - [Ext] DOMAIN_SEPARATOR
  - [Ext] PERMIT_TYPEHASH
  - [Ext] nonces
  - [Ext] permit #
  - [Ext] MINIMUM_LIQUIDITY
  - [Ext] factory
  - [Ext] token0
  - [Ext] token1
  - [Ext] getReserves
  - [Ext] price0CumulativeLast
  - [Ext] price1CumulativeLast
  - [Ext] kLast
  - [Ext] burn #
  - [Ext] swap #
  - [Ext] skim #
  - [Ext] sync #
  - [Ext] initialize #

Smart Contract - Contract Functions

+ [Int] IUniswapV2Router01
  - [Ext] factory
  - [Ext] WETH
  - [Ext] addLiquidity #
  - [Ext] addLiquidityETH ($)
  - [Ext] removeLiquidity #
  - [Ext] removeLiquidityETH #
  - [Ext] removeLiquidityWithPermit #
  - [Ext] removeLiquidityETHWithPermit #
  - [Ext] swapExactTokensForTokens #
  - [Ext] swapTokensForExactTokens #
  - [Ext] swapExactETHForTokens ($)
  - [Ext] swapTokensForExactETH #
  - [Ext] swapExactTokensForETH #
  - [Ext] swapETHForExactTokens ($)
  - [Ext] quote
  - [Ext] getAmountOut
  - [Ext] getAmountIn
  - [Ext] getAmountsOut
  - [Ext] getAmountsIn

+ [Int] IUniswapV2Router02 (IUniswapV2Router01)
  - [Ext] removeLiquidityETHSupportingFeeOnTransferTokens #
  - [Ext] removeLiquidityETHWithPermitSupportingFeeOnTransferTokens #
  - [Ext] swapExactTokensForTokensSupportingFeeOnTransferTokens #
  - [Ext] swapExactETHForTokensSupportingFeeOnTransferTokens ($)
  - [Ext] swapExactTokensForETHSupportingFeeOnTransferTokens #

+ DCM (Context, IERC20, Ownable)
  - [Pub] <Constructor> #
  - [Pub] name
  - [Pub] symbol
  - [Pub] decimals
  - [Pub] totalSupply
  - [Pub] balanceOf
  - [Pub] allowance
  - [Pub] increaseAllowance #
  - [Pub] decreaseAllowance #
  - [Pub] minimumTokensBeforeSwapAmount
  - [Pub] approve #
  - [Prv] _approve #
  - [Pub] setMarketPairStatus #
    - modifiers: onlyOwner
  - [Ext] setIsTxLimitExempt #
    - modifiers: onlyOwner

Smart Contract - Contract Functions

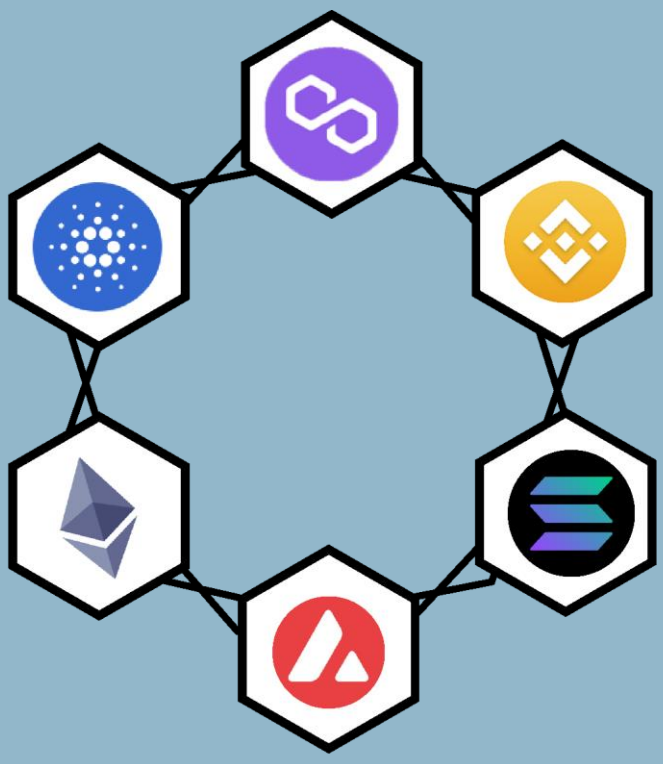
- [Pub] setIsExcludedFromFee #
  - modifiers: onlyOwner
- [Ext] setBuyTaxes #
  - modifiers: onlyOwner
- [Ext] setSellTaxes #
  - modifiers: onlyOwner
- [Ext] setDistributionSettings #
  - modifiers: onlyOwner
- [Ext] setMaxTxAmount #
  - modifiers: onlyOwner
- [Ext] enableDisableWalletLimit #
  - modifiers: onlyOwner
- [Ext] setIsWalletLimitExempt #
  - modifiers: onlyOwner
- [Ext] setWalletLimit #
  - modifiers: onlyOwner
- [Ext] setNumTokensBeforeSwap #
  - modifiers: onlyOwner
- [Ext] setMarketingWalletAddress #
  - modifiers: onlyOwner
- [Pub] setSwapAndLiquifyEnabled #
  - modifiers: onlyOwner
- [Pub] setSwapAndLiquifyByLimitOnly #
  - modifiers: onlyOwner
- [Pub] getCirculatingSupply
- [Prv] transferToAddressETH #
- [Pub] changeRouterVersion #
  - modifiers: onlyOwner
- [Ext] <Receive Ether> ($)
- [Pub] transfer #
- [Pub] transferFrom #
- [Prv] _transfer #
- [Int] _basicTransfer #
- [Prv] swapAndLiquify #
  - modifiers: lockTheSwap
- [Prv] swapTokensForEth #
- [Prv] addLiquidity #
- [Int] takeFee #

Smart Contract - Owner Functions

- Owner can modify buy taxes
- Owner can modify sell taxes
- Owner can modify liquidity and marketing distribution
- Owner can set max tx amount
- Owner can enable/disable wallet limit
- Owner can whitelist addresses from wallet limit
- Owner can set wallet limit
- Owner can set token holdings limit to sell
- Owner can set marketing wallet
- Owner can enable auto liquidity
- Owner can enable auto liquidity when certain limit is set
- Owner can whitelist addresses from max tx limit
- Owner can set AMM pair status

DISCLAIMER

Audits.finance Inc. is in no way responsible or liable for any legal actions resulting from the use of this presentation. By reading this audit or any part of it, you agree to the terms of this disclaimer. If you do not agree to these terms, please stop reading now, and delete any duplicates of this report. Audits.finance Inc. is an official auditor utilizing the Solidity auditing industry standard. Audits.finance hereby excludes any liability and responsibility. Neither you nor any other person shall have any claim against Audits.finance for any economic loss or damages. Audits.finance Inc. does not guarantee the authenticity of a project, nor does it guarantee the project will not participate in one or any scamming including but not limited to, removing liquidity, selling off team supply, or exit scams. Audits.finance Inc. does not give investment advice in any way. Audits.finance Inc. supplies this presentation for information purposes only, and strongly suggests that none of this information be used as investment advice. Audits.finance in no way endorses or recommends any projects that it audits. Audits.finance is solely responsible for smart contract and project analysis of the projects that it is contracted to audit.
Audits.finance may be contracted by teams, investors, or any other 3rd party in regard to a contract address or project. Audits.finance provides a full report for informational purposes only.



audits.finance

Contact information:
Website: audits.finance
Telegram: auditsfinancegroup
Twitter: auditsfinance
Email: hello@audits.finance