AuditAce
FROM INCEPTION TO SUCCESS

Smart Contract Audit

FOR

CEOS

DATED : 07 Mar 23'

AUDIT SUMMARY

Project name – CEOS

Date: 07 March, 2023

Scope of Audit- Audit Ace was consulted to conduct the smart contract audit of the solidity source codes.

Audit Status: **Passed**

Issues Found

Status	Critical	High	Medium	Low	Suggestion
Open	0	0	0	0	0
Acknowledged	0	0	0	0	0
Resolved	0	0	0	0	0

USED TOOLS

Tools:

1- Manual Review:

a line by line code review has been performed by audit ace team.

2- BSC Test Network:

all tests were done on BSC Test network, each test has its transaction has attached to it.

3- Slither : Static Analysis

Testnet Link: all tests were done using this contract, tests are done on BSC Testnet

https://testnet.bscscan.com/token/0xE2453444f6D39f70671B63C7308b7877aF9d086d

Token Information

Token Name : CEOS

Token Symbol: CEOS

Decimals: 9

Token Supply: 420,000,000,000,000,000

Token Address:
0x6f59Da66106480f2033bDAEAcC227803f386c1a0

Checksum:
ea96d3c13d8785eb3d7e143309dddfc842a54fc1

Owner:
0x405c09D2e832e9e99B94267A852E9d24CA9919e9

TOKEN OVERVIEW

Fees:

Buy Fees: 6%

Sell Fees: 6%

Transfer Fees: 0%

Fees Privilige: None

Ownership : Owned

Minting: No mint function

Max Tx Amount/ Max Wallet Amount: No

Blacklist: No

Other Priviliges: including and excluding from fees and rewards

AUDIT METHODOLOGY

The auditing process will follow a routine as special considerations by Auditace:

- Review of the specifications, sources, and instructions provided to Auditace to make sure the contract logic meets the intentions of the client without exposing the user's funds to risk.

- Manual review of the entire codebase by our experts, which is the process of reading source code line-by-line in an attempt to identify potential vulnerabilities.

- Specification comparison is the process of checking whether the code does what the specifications, sources, and instructions provided to Auditace describe.

- Test coverage analysis determines whether the test cases are covering the code and how much code isexercised when we run the test cases.

- Symbolic execution is analysing a program to determine what inputs cause each part of a program to execute.

- Reviewing the codebase to improve maintainability, security, and control based on the established industry and academic practices.

VULNERABILITY CHECKLIST

- ✅ Return values of low-level calls
- ✅ Private modifier
- ✅ Multiple Sends
- ✅ Using Suicide
- ✅ Gas Limitand Loops
- ✅ Address hardcoded
- ✅ Exception Disorder
- ✅ Using inline assembly
- ✅ Divide before multiply
- ✅ Missing Zero Address Validation
- ✅ Compiler version not fixed

- ✅ **Gasless Send**
- ✅ Using block.timestamp
- ✅ Re-entrancy
- ✅ Tautology or contradiction
- ✅ Timestamp Dependence
- ✅ Revert/require functions
- ✅ Use of tx.origin
- ✅ Integer overflow/underflow
- ✅ Dangerous strict equalities
- ✅ Using SHA3
- ✅ Using throw

CLASSIFICATION OF RISK

Severity	Description
◆ **Critical**	These vulnerabilities could be exploited easily and can lead to asset loss, data loss, asset, or data manipulation. They should be fixed right away.
◆ **High-Risk**	A vulnerability that affects the desired outcome when using a contract, or provides the opportunity to use a contract in an unintended way.
◆ **Medium-Risk**	A vulnerability that could affect the desired outcome of executing the contract in a specific scenario.
◆ **Low-Risk**	A vulnerability that does not have a significant impact on possible scenarios for the use of the contract and is probably subjective.
◆ **Gas Optimization /Suggestion**	A vulnerability that has an informational character but is not affecting any of the code.

Findings

Severity	Found
◆ **Critical**	0
◆ **High-Risk**	0
◆ **Medium-Risk**	0
◆ **Low-Risk**	0
◆ **Gas Optimization / Suggestions**	0

INHERITANCE TREE
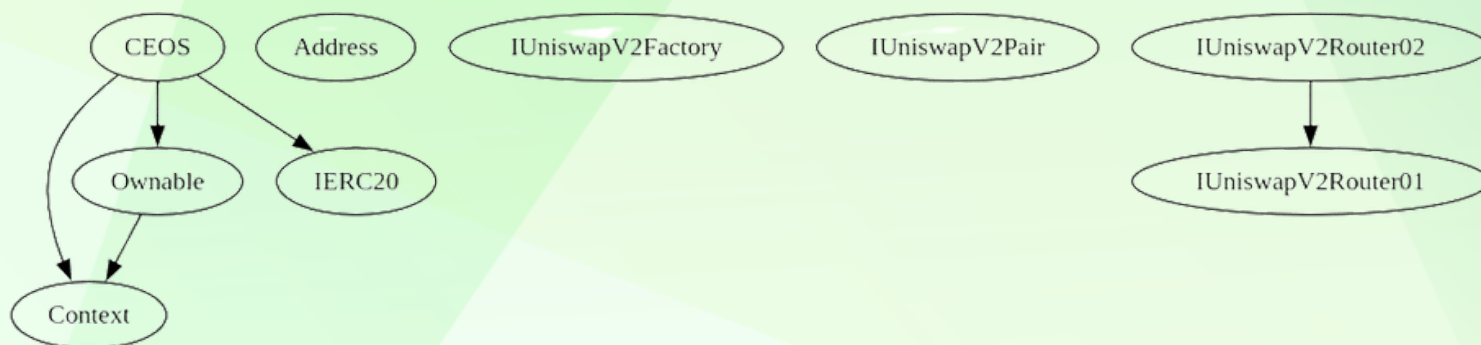
POINTS TO NOTE

- Owner is not able to change fees (6% for buy/sell and 0% for transfers)
- Owner is not able to set max buy/sell/transfer/hold amount
- Owner is not able to blacklist an arbitrary wallet
- Owner is not able to disable trades
- Owner is not able to mint new tokens

CONTRACT ASSESMENT

Contract	Type	Bases		
└	**Function Name**	**Visibility**	**Mutability**	**Modifiers**
Context	Implementation			
└	_msgSender	Internal 🔒		
└	_msgData	Internal 🔒		
Ownable	Implementation	Context		
└	<Constructor>	Public ❗	🛑	NO❗
└	owner	Public ❗		NO❗
└	renounceOwnership	Public ❗	🛑	onlyOwner
└	transferOwnership	Public ❗	🛑	onlyOwner
IERC20	Interface			
└	totalSupply	External ❗		NO❗
└	balanceOf	External ❗		NO❗
└	transfer	External ❗	🛑	NO❗
└	allowance	External ❗		NO❗
└	approve	External ❗	🛑	NO❗
└	transferFrom	External ❗	🛑	NO❗
Address	Library			
└	isContract	Internal 🔒		
└	sendValue	Internal 🔒	🛑	
└	functionCall	Internal 🔒	🛑	
└	functionCall	Internal 🔒	🛑	
└	functionCallWithValue	Internal 🔒	🛑	
└	functionCallWithValue	Internal 🔒	🛑	
└	_functionCallWithValue	Private 🔐	🛑	
IUniswapV2Factory	Interface			
└	feeTo	External ❗		NO❗
└	feeToSetter	External ❗		NO❗
└	getPair	External ❗		NO❗
└	allPairs	External ❗		NO❗
└	allPairsLength	External ❗		NO❗
└	createPair	External ❗	🛑	NO❗
└	setFeeTo	External ❗	🛑	NO❗
└	setFeeToSetter	External ❗	🛑	NO❗
IUniswapV2Pair	Interface			

CONTRACT ASSESMENT

└	name	External ❗			NO❗	
└	symbol	External ❗			NO❗	
└	decimals	External ❗			NO❗	
└	totalSupply	External ❗			NO❗	
└	balanceOf	External ❗			NO❗	
└	allowance	External ❗			NO❗	
└	approve	External ❗		🛑	NO❗	
└	transfer	External ❗		🛑	NO❗	
└	transferFrom	External ❗		🛑	NO❗	
└	DOMAIN_SEPARATOR	External ❗			NO❗	
└	PERMIT_TYPEHASH	External ❗			NO❗	
└	nonces	External ❗			NO❗	
└	permit	External ❗		🛑	NO❗	
└	MINIMUM_LIQUIDITY	External ❗			NO❗	
└	factory	External ❗			NO❗	
└	token0	External ❗			NO❗	
└	token1	External ❗			NO❗	
└	getReserves	External ❗			NO❗	
└	price0CumulativeLast	External ❗			NO❗	
└	price1CumulativeLast	External ❗			NO❗	
└	kLast	External ❗			NO❗	
└	burn	External ❗		🛑	NO❗	
└	swap	External ❗		🛑	NO❗	
└	skim	External ❗		🛑	NO❗	
└	sync	External ❗		🛑	NO❗	
└	initialize	External ❗		🛑	NO❗	
IUniswapV2Router01	Interface			
└	factory	External ❗			NO❗	
└	WETH	External ❗			NO❗	
└	addLiquidity	External ❗		🛑	NO❗	
└	addLiquidityETH	External ❗		💵	NO❗	
└	removeLiquidity	External ❗		🛑	NO❗	
└	removeLiquidityETH	External ❗		🛑	NO❗	
└	removeLiquidityWithPermit	External ❗		🛑	NO❗	
└	removeLiquidityETHWithPermit	External ❗		🛑	NO❗	
└	swapExactTokensForTokens	External ❗		🛑	NO❗	
└	swapTokensForExactTokens	External ❗		🛑	NO❗	
└	swapExactETHForTokens	External ❗		💵	NO❗	
└	swapTokensForExactETH	External ❗		🛑	NO❗	
└	swapExactTokensForETH	External ❗		🛑	NO❗	

CONTRACT ASSESMENT

└	swapETHForExactTokens	External ❗	💵	NO❗
└	quote	External ❗		NO❗
└	getAmountOut	External ❗		NO❗
└	getAmountIn	External ❗		NO❗
└	getAmountsOut	External ❗		NO❗
└	getAmountsIn	External ❗		NO❗
IUniswapV2Router02	Interface	IUniswapV2Router01		
└	removeLiquidityETHSupportingFeeOnTransferTokens	External ❗	🛑	NO❗
└	removeLiquidityETHWithPermitSupportingFeeOnTransferTokens	External ❗	🛑	NO❗
└	swapExactTokensForTokensSupportingFeeOnTransferTokens	External ❗	🛑	NO❗
└	swapExactETHForTokensSupportingFeeOnTransferTokens	External ❗	💵	NO❗
└	swapExactTokensForETHSupportingFeeOnTransferTokens	External ❗	🛑	NO❗
CEOS	Implementation	Context, IERC20, Ownable		
└	<Constructor>	Public ❗	🛑	NO❗
└	name	Public ❗		NO❗
└	symbol	Public ❗		NO❗
└	decimals	Public ❗		NO❗
└	totalSupply	Public ❗		NO❗
└	balanceOf	Public ❗		NO❗
└	transfer	Public ❗	🛑	NO❗
└	allowance	Public ❗		NO❗
└	approve	Public ❗	🛑	NO❗
└	transferFrom	Public ❗	🛑	NO❗
└	increaseAllowance	Public ❗	🛑	NO❗
└	decreaseAllowance	Public ❗	🛑	NO❗
└	isExcludedFromReward	Public ❗		NO❗
└	totalReflectionDistributed	Public ❗		NO❗
└	deliver	Public ❗	🛑	NO❗
└	reflectionFromToken	Public ❗		NO❗
└	tokenFromReflection	Public ❗		NO❗
└	excludeFromReward	Public ❗	🛑	onlyOwner
└	includeInReward	External ❗	🛑	onlyOwner
└	<Receive Ether>	External ❗	💵	NO❗
└	claimStuckTokens	External ❗	🛑	onlyOwner
└	isContract	Internal 🔒		
└	sendBNB	Internal 🔒	🛑	
└	_reflectFee	Private 🔐	🛑	
└	_getValues	Private 🔐		
└	_getTValues	Private 🔐		

CONTRACT ASSESMENT

└	_getRValues	Private 🔓		
└	_getRate	Private 🔓		
└	_getCurrentSupply	Private 🔓		
└	_takeDev	Private 🔓	🛑		
└	_takeMarketing	Private 🔓	🛑		
└	calculateTaxFee	Private 🔓		
└	calculateDevFee	Private 🔓		
└	calculateMarketingFee	Private 🔓		
└	removeAllFee	Private 🔓	🛑		
└	restoreAllFee	Private 🔓	🛑		
└	isExcludedFromFee	Public ❗		NO❗
└	_approve	Private 🔓	🛑		
└	_transfer	Private 🔓	🛑		
└	swapAndSendMarketing	Private 🔓	🛑		
└	swapAndSendDev	Private 🔓	🛑		
└	setSwapTokensAtAmount	External ❗	🛑	onlyOwner
└	setSwapEnabled	External ❗	🛑	onlyOwner
└	_tokenTransfer	Private 🔓	🛑		
└	_transferStandard	Private 🔓	🛑		
└	_transferToExcluded	Private 🔓	🛑		
└	_transferFromExcluded	Private 🔓	🛑		
└	_transferBothExcluded	Private 🔓	🛑		
└	excludeFromFees	External ❗	🛑	onlyOwner
└	changeMarketingWallet	External ❗	🛑	onlyOwner
└	changeDevWallet	External ❗	🛑	onlyOwner

Legend

Symbol	Meaning
🛑	Function can modify state
💵	Function is payable

STATIC ANALYSIS

```
Variable CEOS._transferFromExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1083) is too similar to CEOS._transferFromExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1085)
Variable CEOS._getRValues(uint256,uint256,uint256,uint256,uint256).rTransferAmount (contracts/Token.sol#821) is too similar to CEOS._getTValues(uint256).tTransferAmount (contracts/Token.sol#806)
Variable CEOS._transferToExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1060) is too similar to CEOS._transferBothExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1108)
Variable CEOS._transferBothExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1106) is too similar to CEOS._transferStandard(address,address,uint256).tTransferAmount (contracts/Token.sol#1040)
Variable CEOS._transferBothExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1106) is too similar to CEOS._getValues(uint256).tTransferAmount (contracts/Token.sol#777)
Variable CEOS._getValues(uint256).rTransferAmount (contracts/Token.sol#782) is too similar to CEOS._transferToExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1062)
Variable CEOS._transferToExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1060) is too similar to CEOS._transferStandard(address,address,uint256).tTransferAmount (contracts/Token.sol#1040)
Variable CEOS._getRValues(uint256,uint256,uint256,uint256,uint256).rTransferAmount (contracts/Token.sol#821) is too similar to CEOS._transferStandard(address,address,uint256).tTransferAmount (contracts/Token.sol#1040)
Variable CEOS.reflectionFromToken(uint256,bool).rTransferAmount (contracts/Token.sol#696) is too similar to CEOS._transferStandard(address,address,uint256).tTransferAmount (contracts/Token.sol#1040)
Variable CEOS._getValues(uint256).rTransferAmount (contracts/Token.sol#782) is too similar to CEOS._transferFromExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1085)
Variable CEOS._transferBothExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1106) is too similar to CEOS._getTValues(uint256).tTransferAmount (contracts/Token.sol#806)
Variable CEOS._getValues(uint256).rTransferAmount (contracts/Token.sol#782) is too similar to CEOS._transferBothExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1108)
Variable CEOS._transferFromExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1083) is too similar to CEOS._transferToExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1062)
Variable CEOS._getValues(uint256).rTransferAmount (contracts/Token.sol#782) is too similar to CEOS._transferStandard(address,address,uint256).tTransferAmount (contracts/Token.sol#1040)
Variable CEOS._getValues(uint256).rTransferAmount (contracts/Token.sol#782) is too similar to CEOS._getValues(uint256).tTransferAmount (contracts/Token.sol#777)
Variable CEOS._transferBothExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1106) is too similar to CEOS._transferToExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1062)
Variable CEOS._transferFromExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1083) is too similar to CEOS._transferStandard(address,address,uint256).tTransferAmount (contracts/Token.sol#1040)
Variable CEOS._getValues(uint256).rTransferAmount (contracts/Token.sol#782) is too similar to CEOS._getTValues(uint256).tTransferAmount (contracts/Token.sol#806)
Variable CEOS._transferBothExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1106) is too similar to CEOS._transferFromExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1062)
Variable CEOS._getRValues(uint256,uint256,uint256,uint256,uint256).rTransferAmount (contracts/Token.sol#821) is too similar to CEOS._transferToExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1062)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#variable-names-too-similar

CEOS.setSwapTokensAtAmount(uint256) (contracts/Token.sol#987-993) uses literals with too many digits:
        - require(bool,string)(newAmount > totalSupply() / 100000,SwapTokensAtAmount must be greater than 0.001% of total supply) (contracts/Token.sol#988-991)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#too-many-digits

CEOS.DEAD (contracts/Token.sol#549) is never used in CEOS (contracts/Token.sol#513-1164)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#unused-state-variable

CEOS.DEAD (contracts/Token.sol#549) should be constant
CEOS._decimals (contracts/Token.sol#526) should be constant
CEOS._name (contracts/Token.sol#524) should be constant
CEOS._symbol (contracts/Token.sol#525) should be constant
CEOS.newOwner (contracts/Token.sol#547) should be constant
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-constant

CEOS._tTotal (contracts/Token.sol#529) should be immutable
CEOS.totalFees (contracts/Token.sol#542) should be immutable
CEOS.uniswapV2Pair (contracts/Token.sol#552) should be immutable
CEOS.uniswapV2Router (contracts/Token.sol#551) should be immutable
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-immutable
```

Result => A static analysis of contract's source code has been performed using slither,
No major issues were found in the output

FUNCTIONAL TESTING

Router (PCS V2):
0xD99D1c33F9fC3444f8101754aBC46c52416550D1

All the functionalities have been tested, no issues were found

1- Adding liquidity (passed):
https://testnet.bscscan.com/tx/0x893bd8e4f74277f6dc74a454b6ac2bf41f1b6dea218c72c919b806d3bab50c9b

2- Buying when excluded (0% tax) (passed):
https://testnet.bscscan.com/tx/0xcfb329666d4ad8c0e8760c9c8105687f12f2b9e0f89ff31ee3cc0bce7af0fa96

3- Selling when excluded (0% tax) (passed):
https://testnet.bscscan.com/tx/0x4a1fe96ad831ad393f6e396c9868c972a2182f68fc019d30a0edad3b0c262de3

4- Transferring when transfer without fee enabled (0% tax)
(passed):
https://testnet.bscscan.com/tx/0xb83bf2db8d3aa4c82f5a3fbb674ecf2f62b4766008e37279468aea5b9a12718b

5- Buying when not excluded (10% tax) (passed):
https://testnet.bscscan.com/tx/0xb83bf2db8d3aa4c82f5a3fbb674ecf2f62b4766008e37279468aea5b9a12718b

6- Selling when not excluded (10% tax) (passed):
https://testnet.bscscan.com/tx/0x2d9b134658c9ecd35b185baa3077baa62c662ebf2a4fbf50faea333e62245340

FUNCTIONAL TESTING

7- Transferring when transfer without fee disabled (10% tax) (passed):
https://testnet.bscscan.com/tx/0xdb2bf964eeca9d20169d53f02bb5e5d4f777ce336331c8890e9b106ef5db811b

8- Internal swap (passed):
marketing wallet received ETH
https://testnet.bscscan.com/address/0x2dbe9ffae6d024488bd2af29fb1671fcac2310f6#internaltx

dev wallet received ETH:
https://testnet.bscscan.com/address/0x6a4ee5853425cb9791e95a3914777aa7b8d4cfda

9- Reflections (passed):
we monitors wallet balances for testing this features, wallets received reflection after trades, they stop getting reflections after getting excluded from rewards

MANUAL TESTING

NO ISSUES FOUND

DISCLAIMER

All the content provided in this document is for general information only and should not be used as financial advice or a reason to buy any investment. Team provides no guarantees against the sale of team tokens or the removal of liquidity by the project audited in this document. Always Do your own research and protect yourselves from being scammed. The Auditace team has audited this project for general   information and only expresses their opinion based on similar projects and checks from popular diagnostic tools. Under no circumstances did Auditace receive a payment to manipulate those results or change the awarding badge that we will be adding in our website. Always Do your own research and protect yourselves from scams. This document should not be presented as a reason to buy or not buy any particular token. The Auditace team disclaims any liability for the resulting losses.

ABOUT AUDITACE

We specializes in providing thorough and reliable audits for Web3 projects. With a team of experienced professionals, we use cutting-edge technology and rigorous methodologies to evaluate the security and integrity of blockchain systems. We are committed to helping our clients ensure the safety and transparency of their digital assets and transactions.

https://auditace.tech/

https://t.me/Audit_Ace

https://twitter.com/auditace_

https://github.com/Audit-Ace