



Smart Contract Audit

FOR
Shiba CEO

DATED : 13 MAR 23'



AUDIT SUMMARY

Project name – ShibaCEO

Date: 13 March, 2023

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

Audit Status: **Passed**(Contract developed by Pinksale Safu Dev)

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 Testnet network:

all tests were done on Bsc Testnet 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/0x591c886202597009cC592379B7B88Dd84FAD0eBe#balances>



Token Information

Token Name : Shiba CEO

Token Symbol: ShibaCEO

Decimals: 9

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

Token Address: Not deployed

Checksum:

2d7c64a76f5f92f97d4f45084c7d0c31c350a1a2

Owner: Not Deployed



TOKEN OVERVIEW

Fees:

Buy Fees: 10%

Sell Fees: 10%

Transfer Fees: 10%

Fees Privilege: None

Ownership : Owned

Minting: No mint function

Max Tx Amount/ Max Wallet Amount: No

Blacklist: No

Other Privileges: 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 is exercised 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 |  Gasless Send |
|  Private modifier |  Using block.timestamp |
|  Multiple Sends |  Re-entrancy |
|  Using Suicide |  Tautology or contradiction |
|  Gas Limitand Loops |  Timestamp Dependence |
|  Address hardcoded |  Revert/require functions |
|  Exception Disorder |  Use of tx.origin |
|  Using inline assembly |  Integer overflow/underflow |
|  Divide before multiply |  Dangerous strict equalities |
|  Missing Zero Address Validation |  Using SHA3 |
|  Compiler version not fixed |  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

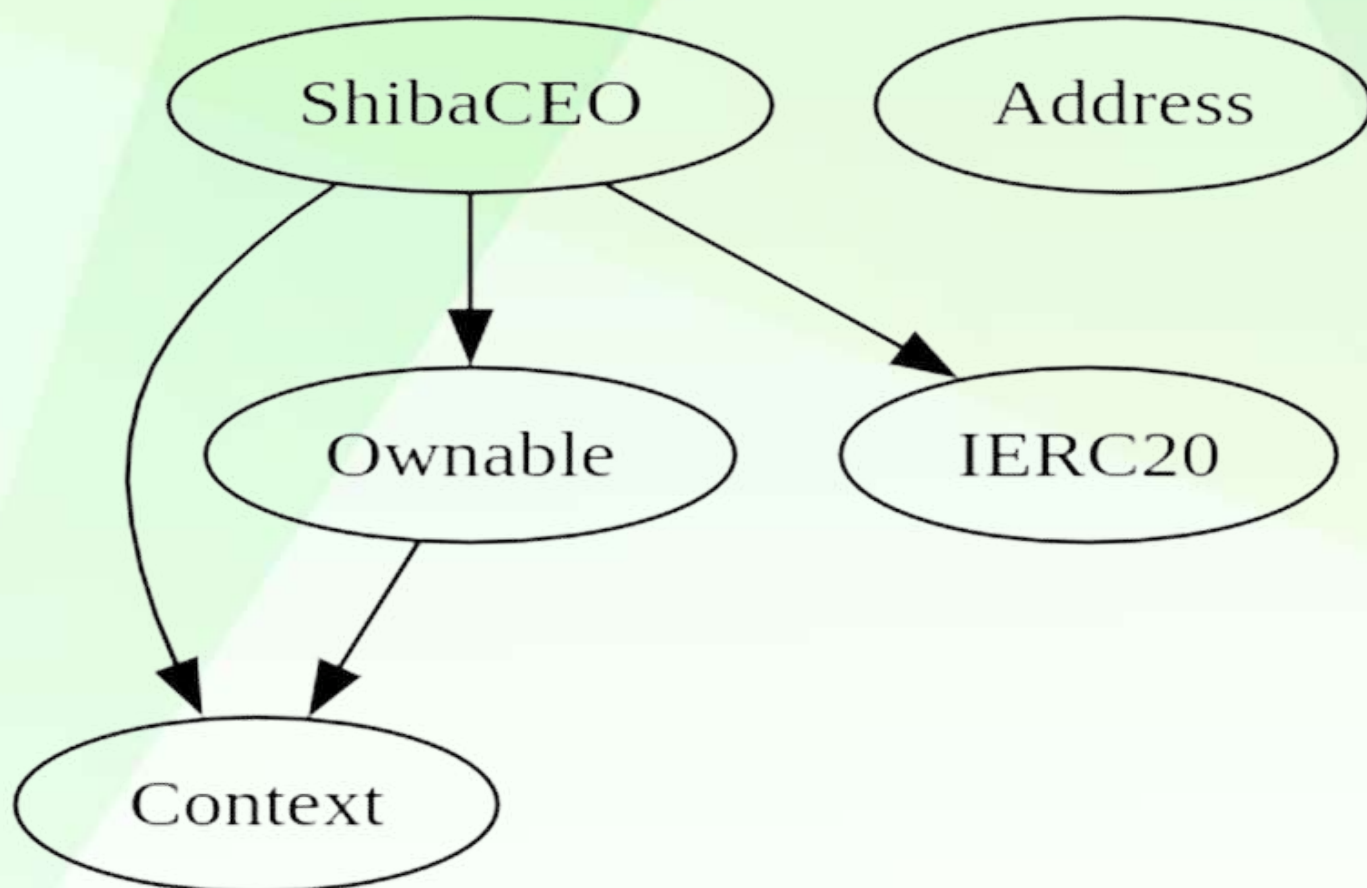
◆ Low-Risk

0

◆ Gas Optimization / Suggestions

0

INHERITANCE TREE

























POINTS TO NOTE

- **Owner is not able to change fees (10% for buy/sell/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 ! |
|  |  |
| **ShibaCEO** | 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 |
|  | _reflectFee | Private  |  |  |
|  | _getValues | Private  |  |  |
|  | _getTValues | Private  |  |  |
|  | _getRValues | Private  |  |  |
|  | _getRate | Private  |  |  |
```

CONTRACT ASSESMENT

	└		_getCurrentSupply		Private	🔒			
	└		_takeLiquidity		Private	🔒		🛑	
	└		_takeMarketing		Private	🔒		🛑	
	└		calculateTaxFee		Private	🔒			
	└		calculateLiquidityFee		Private	🔒			
	└		calculateMarketingFee		Private	🔒			
	└		removeAllFee		Private	🔒		🛑	
	└		setBuyFee		Private	🔒		🛑	
	└		setSellFee		Private	🔒		🛑	
	└		isExcludedFromFee		Public	!		NO!	
	└		_approve		Private	🔒		🛑	
	└		_transfer		Private	🔒		🛑	
	└		swapAndLiquify		Private	🔒		🛑	
	└		swapAndSendMarketing		Private	🔒		🛑	
	└		setSwapTokensAtAmount		External	!		🛑	
	└		setSwapEnabled		External	!		🛑	
	└		_tokenTransfer		Private	🔒		🛑	
	└		_transferStandard		Private	🔒		🛑	
	└		_transferToExcluded		Private	🔒		🛑	
	└		_transferFromExcluded		Private	🔒		🛑	
	└		_transferBothExcluded		Private	🔒		🛑	
	└		excludeFromFees		External	!		🛑	

Legend

	Symbol		Meaning	
	:-----:		-----	
	🛑		Function can modify state	
	💰		Function is payable	



STATIC ANALYSIS

```
Variable ShibaCEO._transferToExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1139) is too similar to ShibaCEO._transferToExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1141)
Variable ShibaCEO._getRValues(uint256,uint256,uint256,uint256).rTransferAmount (contracts/Token.sol#868) is too similar to ShibaCEO._transferFromExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1164)
Variable ShibaCEO._transferToExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1139) is too similar to ShibaCEO._getValues(uint256).tTransferAmount (contracts/Token.sol#824)
Variable ShibaCEO.reflectionFromToken(uint256,bool).rTransferAmount (contracts/Token.sol#758) is too similar to ShibaCEO._getValues(uint256).tTransferAmount (contracts/Token.sol#824)
Variable ShibaCEO._transferBothExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1183) is too similar to ShibaCEO._transferToExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1141)
Variable ShibaCEO.reflectionFromToken(uint256,bool).rTransferAmount (contracts/Token.sol#758) is too similar to ShibaCEO._transferToExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1141)
Variable ShibaCEO._transferStandard(address,address,uint256).rTransferAmount (contracts/Token.sol#1117) is too similar to ShibaCEO._transferFromExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1164)
Variable ShibaCEO._transferStandard(address,address,uint256).rTransferAmount (contracts/Token.sol#1117) is too similar to ShibaCEO._getValues(uint256).tTransferAmount (contracts/Token.sol#853)
Variable ShibaCEO._transferStandard(address,address,uint256).rTransferAmount (contracts/Token.sol#1117) is too similar to ShibaCEO._transferBothExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1187)
Variable ShibaCEO._transferToExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1139) is too similar to ShibaCEO._transferFromExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1164)
Variable ShibaCEO._transferToExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1139) is too similar to ShibaCEO._getValues(uint256).tTransferAmount (contracts/Token.sol#853)
Variable ShibaCEO._transferFromExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1162) is too similar to ShibaCEO._transferToExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1141)
Variable ShibaCEO._getRValues(uint256,uint256,uint256,uint256).rTransferAmount (contracts/Token.sol#868) is too similar to ShibaCEO._transferToExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1141)
Variable ShibaCEO.reflectionFromToken(uint256,bool).rTransferAmount (contracts/Token.sol#758) is too similar to ShibaCEO._transferFromExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1164)
Variable ShibaCEO._transferBothExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1183) is too similar to ShibaCEO._transferFromExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1164)
Variable ShibaCEO._transferToExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1139) is too similar to ShibaCEO._transferBothExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1187)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#variable-names-too-similar

ShibaCEO.DEAD (contracts/Token.sol#565) should be constant
ShibaCEO.decimals (contracts/Token.sol#539) should be constant
ShibaCEO.name (contracts/Token.sol#537) should be constant
ShibaCEO.symbol (contracts/Token.sol#538) should be constant
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-constant

ShibaCEO.tTotal (contracts/Token.sol#542) should be immutable
ShibaCEO.liquidityFeeonBuy (contracts/Token.sol#549) should be immutable
ShibaCEO.liquidityFeeonSell (contracts/Token.sol#550) should be immutable
ShibaCEO.marketingFeeonBuy (contracts/Token.sol#552) should be immutable
ShibaCEO.marketingFeeonSell (contracts/Token.sol#553) should be immutable
ShibaCEO.marketingWallet (contracts/Token.sol#562) should be immutable
ShibaCEO.marketingWalletTwo (contracts/Token.sol#563) should be immutable
ShibaCEO.taxFeeonBuy (contracts/Token.sol#546) should be immutable
ShibaCEO.taxFeeonSell (contracts/Token.sol#547) should be immutable
ShibaCEO.totalBuyFees (contracts/Token.sol#559) should be immutable
ShibaCEO.totalSellFees (contracts/Token.sol#560) should be immutable
ShibaCEO.uniswapV2Pair (contracts/Token.sol#568) should be immutable
ShibaCEO.uniswapV2Router (contracts/Token.sol#567) 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 issues found



FUNCTIONAL TESTING

Router (PCS V2):

0xD99D1c33F9fC3444f8101754aBC46c52416550D1

1- Adding Liquidity (Passed):

liquidity added on Pancakeswap V2:

<https://testnet.bscscan.com/tx/0x7c83426bed83de613569cfeed38488ce0d6df681b114fd6993e8c98c5ac93a75>

2- Buying from a non-excluded wallet (0% max)(Passed):

<https://testnet.bscscan.com/tx/0x3b7ba9d6952863fcdeed8f0f9c22e17e42f4f4e7a5a8334a55f69c1a6c2abeb5>

3- Selling from a non-excluded wallet (0% max)(Passed):

<https://testnet.bscscan.com/tx/0xf8a6a6960fae0596034f11b56e4e7db65cadcc9c5071d56c010abdee196abb2c>

4- Transferring when excluded (0% tax) (passed):

<https://testnet.bscscan.com/tx/0x9e4ed6daef99ee3d0c56a2a28037a2b01922a48b7751c3c57665e3dbfaeb0d8b>

5- Buying when not excluded (10% tax) (passed):

<https://testnet.bscscan.com/tx/0x170d2edb479f4867dcbcb5d153a3d3385907cd6bd3e4acd864799d1fc413f9f6a>



FUNCTIONAL TESTING

6- Selling when not excluded (10% tax) (**passed**):

<https://testnet.bscscan.com/tx/0x21b640958889a18fbe50fe0f7422936ac8559317b1d3d730f0c46292c22685f6>

7- Transferring when not excluded(10% tax) (**passed**):

<https://testnet.bscscan.com/tx/0x9e4ed6daef99ee3d0c56a2a28037a2b01922a48b7751c3c57665e3dbfaeb0d8b>

8- Internal swap (**passed**):

marketing wallet 2 received ETH

<https://testnet.bscscan.com/address/0x3cde8c587e658c4275a27212b943c865edf1618e#internaltx>

marketing wallet 1 received ETH

<https://testnet.bscscan.com/address/0xc6f92e3f0d46d5418e400c55b97cfb5364f3b3af#internaltx>

9- Reflections (**passed**):

we monitored wallet balances for testing this features, wallets received reflection after trades, they stoped getting reflections after getting excluded from rewards



MANUAL TESTING

No Issues Found



Social Media Overview

**Here are the Social Media Accounts of
Shiba CEO**



<https://t.me/ShibaCEOEN>



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>
