COINSCOPE

Audit Report

Cartoon Doge

February 2022

Type	BEP20
Network	BSC
Address	0xeb58B47404F386286b0BF09cAE3C14d8b82F65CF
Audited by	© coinscope

Table of Contents

Table of Contents 1

Contract Review 3

Audit Updates 3

Contract Analysis 4

ELFM - Exceed Limit Fees Manipulation 5

Description 5

Recommendation 6

Staking Balance Calculation 7

Stacking and Users Balance Inconsistency 7

Contract Diagnostics 8

L01 - Public Function could be Declared External 9

Description 9

Recommendation 9

L02 - State Variables could be Declared Constant 10

Description 10

Recommendation 10

L04 - Conformance to Solidity Naming Conventions 11

Description 11

Recommendation 11

L09 - Dead Code Elimination 12

Description 12

Recommendation 12

L07 - Missing Events Arithmetic 13

Description 13

Recommendation 13

L15 - Local Scope Variable Shadowing 14

Description 14
Recommendation 14
L13 - Divide before Multiply Operation 15
Description 15
Recommendation 15
Contract Functions 16
Contract Flow 22
Domain Info 23
Summary 24
Disclaimer 25
About Coinscope 26

Contract Review

Contract Name	Cartoondoge
Compiler Version	v0.8.4+commit.c7e474f2
Optimization	200 runs
Licence	MIT
Explorer	https://bscscan.com/token/0xeb58B47404F386286b0BF09cAE3C14d8b82F65CF
Symbol	CARTD
Decimals	18
Total Supply	957,019,432,195,840
Source	contract.sol
Domain	cartoondoge.com

Audit Updates

Initial Audit	17th February 2022
Corrected	

Contract Analysis

🔴 Critical 🟠 Medium ⚪ Minor 🔵 Pass

Severity	Code	Description
🔵	ST	Contract Owner is not able to stop or pause transactions
🔵	OCTD	Contract Owner is not able to transfer tokens from specific address
🔵	OTUT	Owner Transfer User's Tokens
🟠	ELFM	Contract Owner is not able to increase fees more than a reasonable percent (25%)
🔵	ULTW	Contract Owner is not able to increase the amount of liquidity taken by dev wallet more than a reasonable percent
🔵	MT	Contract Owner is not able to mint new tokens
🔵	BT	Contract Owner is not able to burn tokens from specific wallet
🔵	BC	Contract Owner is not able to blacklist wallets from selling

ELFM - Exceed Limit Fees Manipulation

Criticality	medium
Location	contract.sol#L1066

Description

The contract owner has the authority to increase over the allowed limit of 25%. The owner may take advantage of it by calling the `UpdateTaxes` function with a high percentage value on the transferTax argument.

```
function UpdateTaxes(uint8 burnTaxes, uint8 buybackTaxes, uint8 devTaxes, uint8
marketingTaxes, uint8 liquidityTaxes, uint8 stakingTaxes,uint8 buyTax, uint8
sellTax, uint8 transferTax) public authorized{
    uint8
totalTax=liquidityTaxes+stakingTaxes+marketingTaxes+burnTaxes+buybackTaxes+devTa
xes;

    //buy and sell tax can never be higher than MaxTax set at beginning of
contract
    //this prevents owner from setting ridiculous tax or turning contract into
honeypot
    require(totalTax==100, "marketing+liq+staking needs to equal 100%");
    require(buyTax<=MaxTax&&sellTax<=MaxTax,"taxes higher than max tax");
    require(transferTax<=50,"transferTax higher than max transferTax");
    _burnTax=burnTaxes;
    _buyBackTax=buybackTaxes;
    _devTax=devTaxes;
    _marketingTax=marketingTaxes;
    _liquidityTax=liquidityTaxes;
    _stakingTax=stakingTaxes;
    _buyTax=buyTax;
    _sellTax=sellTax;
    _transferTax=transferTax;
    emit
OnUpdateTaxes(burnTaxes,buybackTaxes,devTaxes,marketingTaxes,liquidityTaxes,stak
ingTaxes,buyTax,sellTax,transferTax);
}
```

Recommendation

The contract could embody a check for the maximum acceptable value.

The team should carefully manage the private keys of the owner’s account. We strongly recommend a powerful security mechanism that will prevent a single user from accessing the contract admin functions. That risk can be prevented by temporarily locking the contract or renouncing ownership.

Staking Balance Calculation

The staking reward amount is calculated from the swap and liquify feature.
The contract swaps the tokens for BNB. The first portion is added to the liquidity and the remaining is added to the stacking reward amount.
The users have the ability to claim their reward in tokens instead of BNB. As a result, there is an amount in BNB that is accumulated to the contract.

Stacking and Users Balance Inconsistency

The contract is aiming to keep the user's staking amount proportional to the user's balance. There are many segments where this procedure is taking place. There is no clear pattern that keeps these two arrays updated. Hence, the contract is vulnerable on reentrancy attacks.

Contract Diagnostics

● Critical ● Medium ● Minor

Severity	Code	Description
●	L01	Public Function could be Declared External
●	L02	State Variables could be Declared Constant
●	L04	Conformance to Solidity Naming Conventions
●	L09	Dead Code Elimination
●	L07	Missing Events Arithmetic
●	L15	Local Scope Variable Shadowing
●	L13	Divide before Multiply Operation

L01 - Public Function could be Declared External

Criticality	minor
Location	contract.sol#L209,213,547,552,598,604,611,619,631,642 and 24 more

Description

Public functions that are never called by the contract should be declared external to save gas.

```
getTaxes
getLiquidityUnlockInSeconds
getLimits
UpdateLimits
UpdateRewardSplit
UpdateTaxes
IncludeAccountToFees
ExcludeAccountFromFees
SetDevWallet
...
```

Recommendation

Use the external attribute for functions never called from the contract

L02 - State Variables could be Declared Constant

Criticality	minor
Location	contract.sol#L406

Description

Constant state variables should be declared constant to save gas.

```
BurnAddress
```

Recommendation

Add the constant attribute to state variables that never change.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contract.sol#L48,49,148,694,949,453,547,552,611,619 and 49 more

Description

Solidity defines a naming convention that should be followed. Rule exceptions:

- Allow constant variable name/symbol/decimals to be lowercase.
- Allow _ at the beginning of the mixed_case match for private variables and unused parameters.

```
AutoLPThreshold
BuyBackBalance
DevBalance
MarketingBalance
DistributionMultiplier
MiscReward
MainReward
PancakeRouter
BurnAddress
...
```

Recommendation

Follow the Solidity naming convention.
https://docs.soliditylang.org/en/v0.4.25/style-guide.html#naming-conventions

L09 - Dead Code Elimination

Criticality	minor
Location	contract.sol#L261,230,233,236,239,253,256,245,248,220 and 15 more

Description

Functions that are not used in the contract, and make the code's size bigger.

```
remove
length
contains
at
add
_length
_at
sendValue
isContract
...
```

Recommendation

Remove unused functions.

L07 - Missing Events Arithmetic

Criticality	minor
Location	contract.sol#L970,986

Description

Detected missing events for critical arithmetic parameters. There are functions that have no event emitted, so it is difficult to track off-chain changes.

```
MarketingBalance -= amount
DevBalance -= amount
```

Recommendation

Emit an event for critical parameter changes.

L15 - Local Scope Variable Shadowing

Criticality	minor
Location	contract.sol#L453,460

Description

The are variables that are defined in the local scope containing the same name from an upper scope.

```
owner
_owner
```

Recommendation

The local variables should have different names from the upper scoped variables.

L13 - Divide before Multiply Operation

Criticality	minor
Location	contract.sol#L762,794

Description

Performing divisions before multiplications may cause lose of prediction.

```
tokenToSwap = _balances[_pancakePairAddress] * permilleOfPancake / 1000
MiscAmount = (amount * _MiscRewardSplit) / 100
MainAmount = (amount * _MainRewardSplit) / 100
```

Recommendation

The multiplications should be prior to the divisions.

Contract Functions

Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
IBEP20	Interface			
	totalSupply	External		-
	decimals	External		-
	symbol	External		-
	name	External		-
	getOwner	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
IPancakeERC20	Interface			
	name	External		-
	symbol	External		-
	decimals	External		-
	totalSupply	External		-
	balanceOf	External		-
	allowance	External		-
	approve	External	✓	-
	transfer	External	✓	-
	transferFrom	External	✓	-
	DOMAIN_SEPARATOR	External		-
	PERMIT_TYPEHASH	External		-
	nonces	External		-
	permit	External	✓	-
IPancakeFacto	Interface			

ry				
	feeTo	External		-
	feeToSetter	External		-
	getPair	External		-
	allPairs	External		-
	allPairsLength	External		-
	createPair	External	✓	-
	setFeeTo	External	✓	-
	setFeeToSetter	External	✓	-
IPancakeRouter01	Interface			
	addLiquidity	External	✓	-
	addLiquidityETH	External	Payable	-
	removeLiquidity	External	✓	-
	removeLiquidityETH	External	✓	-
	removeLiquidityWithPermit	External	✓	-
	removeLiquidityETHWithPermit	External	✓	-
	swapExactTokensForTokens	External	✓	-
	swapTokensForExactTokens	External	✓	-
	swapExactETHForTokens	External	Payable	-
	swapTokensForExactETH	External	✓	-
	swapExactTokensForETH	External	✓	-
	swapETHForExactTokens	External	Payable	-
	factory	External		-
	WETH	External		-
	quote	External		-
	getamountOut	External		-
	getamountIn	External		-
	getamountsOut	External		-
	getamountsIn	External		-
IPancakeRouter02	Interface	IPancakeRouter01		
	removeLiquidityETHSupportingFeeOnTransferTokens	External	✓	-

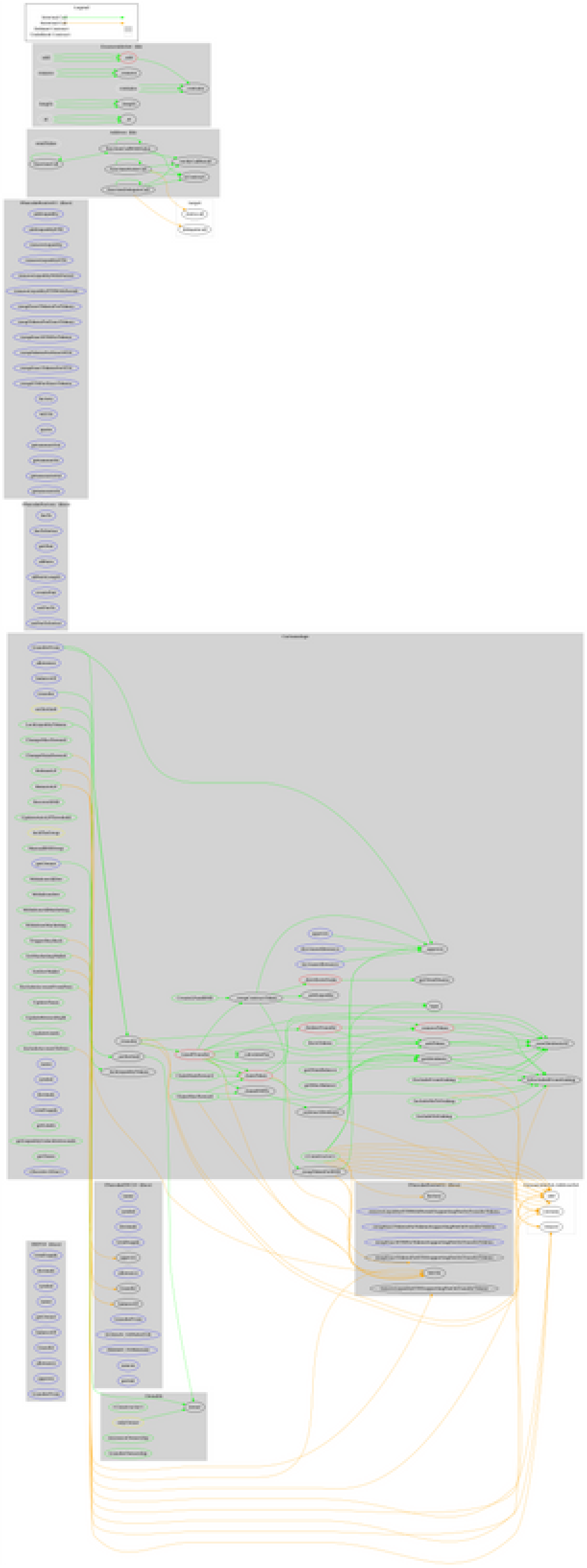
	removeLiquidityETHWithPermitSupportingFeeOnTransferTokens	External	✓	-
	swapExactTokensForTokensSupportingFeeOnTransferTokens	External	✓	-
	swapExactETHForTokensSupportingFeeOnTransferTokens	External	Payable	-
	swapExactTokensForETHSupportingFeeOnTransferTokens	External	✓	-
Ownable	Implementation			
	<Constructor>	Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
Address	Library			
	isContract	Internal		
	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionStaticCall	Internal		
	functionStaticCall	Internal		
	functionDelegateCall	Internal	✓	
	functionDelegateCall	Internal	✓	
	_verifyCallResult	Private		
EnumerableSet	Library			
	_add	Private	✓	
	_remove	Private	✓	
	_contains	Private		
	_length	Private		
	_at	Private		
	add	Internal	✓	
	remove	Internal	✓	

	contains	Internal		
	length	Internal		
	at	Internal		
	add	Internal	✓	
	remove	Internal	✓	
	contains	Internal		
	length	Internal		
	at	Internal		
	add	Internal	✓	
	remove	Internal	✓	
	contains	Internal		
	length	Internal		
	at	Internal		
Cartoondoge	Implementation	IBEP20, Ownable		
	_authorized	Private		
	<Constructor>	Public	✓	-
	allowance	External		-
	approve	External	✓	-
	_approve	Private	✓	
	balanceOf	External		-
	decreaseAllowance	External	✓	-
	increaseAllowance	External	✓	-
	transfer	External	✓	-
	transferFrom	External	✓	-
	claimToken	Private	✓	
	ClaimMainReward	Public	✓	-
	ClaimMiscReward	Public	✓	-
	_claimBNBTo	Private	✓	
	_subtractDividents	Private	✓	
	getDividents	Private		
	getMainBalance	Public		-
	getMiscBalance	Public		-
	ChangeMainReward	Public	✓	authorized
	ChangeMiscReward	Public	✓	authorized

	LockLiquidityTokens	Public	✓	onlyOwner
	_lockLiquidityTokens	Private	✓	
	ReleaseLP	Public	✓	onlyOwner
	RemoveLP	Public	✓	onlyOwner
	RecoverBNB	Public	✓	onlyOwner
	UpdateAutoLPThreshold	Public	✓	authorized
	CreateLPandBNB	Public	✓	authorized
	getTotalShares	Public		-
	isExcludedFromStaking	Public		-
	_addToken	Private	✓	
	_removeToken	Private	✓	
	_newDividentsOf	Private		
	_distributeStake	Private	✓	
	_swapContractToken	Private	✓	lockTheSwap
	_swapTokenForBNB	Private	✓	
	_addLiquidity	Private	✓	
	ManualBNBSwap	Public	✓	authorized
	_calculateFee	Private		
	_feelessTransfer	Private	✓	
	_taxedTransfer	Private	✓	
	_transfer	Private	✓	
	TriggerBuyBack	Public	✓	authorized
	BurnTokens	Public	✓	-
	WithdrawAllDev	Public	✓	authorized
	WithdrawDev	Public	✓	authorized
	WithdrawAllMarketing	Public	✓	authorized
	WithdrawMarketing	Public	✓	authorized
	ExcludeFromStaking	Public	✓	authorized
	IncludeMeToStaking	Public	✓	-
	IncludeToStaking	Public	✓	authorized
	SetMarketingWallet	Public	✓	onlyOwner
	SetDevWallet	Public	✓	onlyOwner
	ExcludeAccountFromFees	Public	✓	authorized
	IncludeAccountToFees	Public	✓	authorized
	UpdateTaxes	Public	✓	authorized

	UpdateRewardSplit	Public	✓	authorized
	UpdateLimits	Public	✓	authorized
	getOwner	External		-
	name	External		-
	symbol	External		-
	decimals	External		-
	totalSupply	External		-
	getLimits	Public		-
	getLiquidityUnlockInSeconds	Public		-
	getTaxes	Public		-
	<Receive Ether>	External	Payable	-

Contract Flow



Domain Info

Domain Name	cartoondoge.com
Registry Domain ID	2656072831_DOMAIN_COM-VRSN
Creation Date	2021-11-19T14:03:07Z
Updated Date	2021-11-19T14:03:08Z
Registry Expiry Date	2022-11-19T14:03:07Z
Registrar WHOIS Server	whois.wix.com
Registrar URL	http://www.wix.com
Registrar	Wix.com Ltd.
Registrar IANA ID	3817

The domain has been created 3 months before the creation of the audit. It will expire in 9 months.

There is no public billing information, the creator is protected by the privacy settings.

Summary

The contract owner has the ability to manipulate the fees over the allowed limit. The business logic of the contract may produce some inconsistency between the expected and the actual state. The audit mentions the security concerns, business logic notes and performance improvements.

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.

Coinscope 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 Coinscope 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 Coinscope 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 Coinscope team disclaims any liability for the resulting losses.

About Coinscope

Coinscope is the leading early coin listing, voting and auditing authority firm. The audit process is analyzing and monitoring many aspects of the project. That way, it gives the community a good sense of security using an informative report and a generic score.

Coinscope is aiming to make crypto discoverable and efficient globally. It provides all the essential tools to assist users draw their own conclusions.

The Coinscope.co team

https://www.coinscope.co