



# Audit Report

# **Community Support**

December 2021

Type           BEP20 Testnet

Address       0x6584e787cF7a6a465a6b8cD1c968EcA8AbB9c071

Audited by   © coinscope

# Table of Contents

<b>Table of Contents</b>	<b>1</b>
<b>Contract Review</b>	<b>3</b>
<b>Audit Updates</b>	<b>3</b>
<b>Contract Analysis</b>	<b>4</b>
<b>ST - Stop Transactions</b>	<b>4</b>
Description	5
Recommendation	5
<b>ELFM - Exceed Limit Fees Manipulation</b>	<b>6</b>
Description	6
Recommendation	6
<b>Contract Diagnostics</b>	<b>8</b>
<b>L01 - Public Function could be Declared External</b>	<b>9</b>
Description	9
Recommendation	9
<b>L02 - State Variables could be Declared Constant</b>	<b>10</b>
Description	10
Recommendation	10
<b>L03 - Redundant Statements</b>	<b>11</b>
Description	11
Recommendation	11
<b>L04 - Conformance to Solidity Naming Conventions</b>	<b>12</b>
Description	12
Recommendation	12
<b>L09 - Dead Code Elimination</b>	<b>13</b>
Description	13
Recommendation	13

<b>L07 - Missing Events Arithmetic</b>	<b>14</b>
<b>Description</b>	<b>14</b>
<b>Recommendation</b>	<b>14</b>
<b>Contract Functions</b>	<b>15</b>
<b>Contract Flow</b>	<b>21</b>
<b>Summary</b>	<b>22</b>
<b>Disclaimer</b>	<b>23</b>
<b>About Coinscope</b>	<b>24</b>

## Contract Review

<b>Contract Name</b>	CST
<b>Compiler Version</b>	v0.8.7+commit.e28d00a7
<b>Optimization</b>	200 runs
<b>Licence</b>	MIT
<b>Explorer</b>	<a href="https://testnet.bscscan.com/token/0x6584e787cf7a6a465a6b8cd1c968eca8abb9c071">https://testnet.bscscan.com/token/0x6584e787cf7a6a465a6b8cd1c968eca8abb9c071</a>
<b>Symbol</b>	CST
<b>Decimals</b>	9
<b>Total Supply</b>	1,000,000,000,000,000
<b>Website</b>	

## Audit Updates

<b>Initial Audit</b>	31th of December 2021
<b>Corrected</b>	

# Contract Analysis

● Critical
 ● Medium
 ● Minor
 ● Pass

Severity	Code	Description
<span style="color: gold;">●</span>	ST	Contract Owner is not able to stop or pause transactions
<span style="color: blue;">●</span>	OCTD	Contract Owner is not able to transfer tokens from specific address
<span style="color: blue;">●</span>	OTUT	Owner Transfer User's Tokens
<span style="color: red;">●</span>	ELFM	Contract Owner is not able to increase fees more than a reasonable percent (25%)
<span style="color: blue;">●</span>	ULTW	Contract Owner is not able to increase the amount of liquidity taken by dev wallet more than a reasonable percent
<span style="color: blue;">●</span>	MT	Contract Owner is not able to mint new tokens
<span style="color: blue;">●</span>	BT	Contract Owner is not able to burn tokens from specific wallet
<span style="color: blue;">●</span>	BC	Contract Owner is not able to blacklist wallets from selling

## ST - Stop Transactions

<b>Criticality</b>	medium
<b>Location</b>	contract.sol#L656

### Description

The contract owner has the authority to stop transactions for all users excluding the owner. The owner may take advantage of it by setting the `_maxTxAmount` to zero.

```
if(from != owner() && to != owner())
{
    require(amount <= _maxTxAmount, "Transfer amount exceeds the maxTxAmount.");
}
```

### Recommendation

The contract could embody a check for not allowing setting the `_maxTxAmount` less than a reasonable amount. A suggested implementation could check that the maximum amount should be more than a fixed percentage of the total supply.

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.

## ELFM - Exceed Limit Fees Manipulation

<b>Criticality</b>	critical
<b>Location</b>	contract.sol#L904

### 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 `setAllFeesPercent` function with a high percentage value.

```
function setAllFeesPercent(  
    uint256 taxFee,  
    uint256 liquidityFee,  
    uint256 marketingFee,  
    uint256 developerFee,  
    uint256 burnFee  
) public onlyOwner() {  
    _taxFee = taxFee;  
    _previousTaxFee = _taxFee;  
  
    _liquidityFee = liquidityFee;  
    _previousLiquidityFee = _liquidityFee;  
  
    _marketingFee = marketingFee;  
    _previousMarketingFee = _marketingFee;  
  
    _developerFee = developerFee;  
    _previousDeveloperFee = _developerFee;  
  
    _burnFee = burnFee;  
    _previousBurnFee = _burnFee;  
}
```

### 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.





# Contract Diagnostics

● Critical    ● Medium    ● Minor

Severity	Code	Description
●	L01	Public Function could be Declared External
●	L02	State Variables could be Declared Constant
●	L03	Redundant Statements
●	L04	Conformance to Solidity Naming Conventions
●	L09	Dead Code Elimination
●	L07	Missing Events Arithmetic

## L01 - Public Function could be Declared External

<b>Criticality</b>	minor
<b>Location</b>	contract.sol#L1027,L1022,L1018 and 27 more

### Description

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

```
setMaxWalletLimit  
setExcludedFromWhale  
setExtraAntiWhaleFee  
...
```

### Recommendation

Use the external attribute for functions never called from the contract

## L02 - State Variables could be Declared Constant

**Criticality**

minor

**Location**

contract.sol#L441,L453,L458 and 2 more

### Description

Constant state variables should be declared constant to save gas.

```
developerAddress  
_tTotal  
_symbol  
...
```

### Recommendation

Add the constant attribute to state variables that never change.

## L03 - Redundant Statements

**Criticality**

minor

**Location**

contract.sol#L10

### Description

Detect the usage of redundant statements that have no effect.

Context

### Recommendation

Remove redundant statements if they congest code but offer no value.

## L04 - Conformance to Solidity Naming Conventions

**Criticality**

minor

**Location**

contract.sol#L487,L485,L483 and 22 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.

```
_extraAntiWhaleFee  
_maxSaleAmount  
_maxTxAmount  
...
```

### 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#L74,L70,L9 and 7 more

### Description

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

```
mod
_msgData
sendValue
...
```

### Recommendation

Remove unused functions.

## L07 - Missing Events Arithmetic

<b>Criticality</b>	minor
<b>Location</b>	contract.sol#L1027,L1018,L965 and 4 more

### 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.

```
maxLimit = amount
_extraAntiWhaleFee = _value
minimumTokensBeforeSwap = _minimumTokensBeforeSwap
...
```

### Recommendation

Emit an event for critical parameter changes.

# Contract Functions

Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
<b>Context</b>	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
<b>IBEP20</b>	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
<b>SafeMath</b>	Library			
	add	Internal		
	sub	Internal		
	sub	Internal		
	mul	Internal		
	div	Internal		
	div	Internal		
	mod	Internal		
	mod	Internal		
<b>Address</b>	Library			
	isContract	Internal		
	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	Internal	✓	



	_functionCallWithValue	Private	✓	
<b>Ownable</b>	Implementation	Context		
	<Constructor>	Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
	getUnlockTime	Public		-
	getTime	Public		-
	lock	Public	✓	onlyOwner
	unlock	Public	✓	-
<b>IUniswapV2Factory</b>	Interface			
	feeTo	External		-
	feeToSetter	External		-
	getPair	External		-
	allPairs	External		-
	allPairsLength	External		-
	createPair	External	✓	-
	setFeeTo	External	✓	-
	setFeeToSetter	External	✓	-
<b>IUniswapV2Pair</b>	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		-

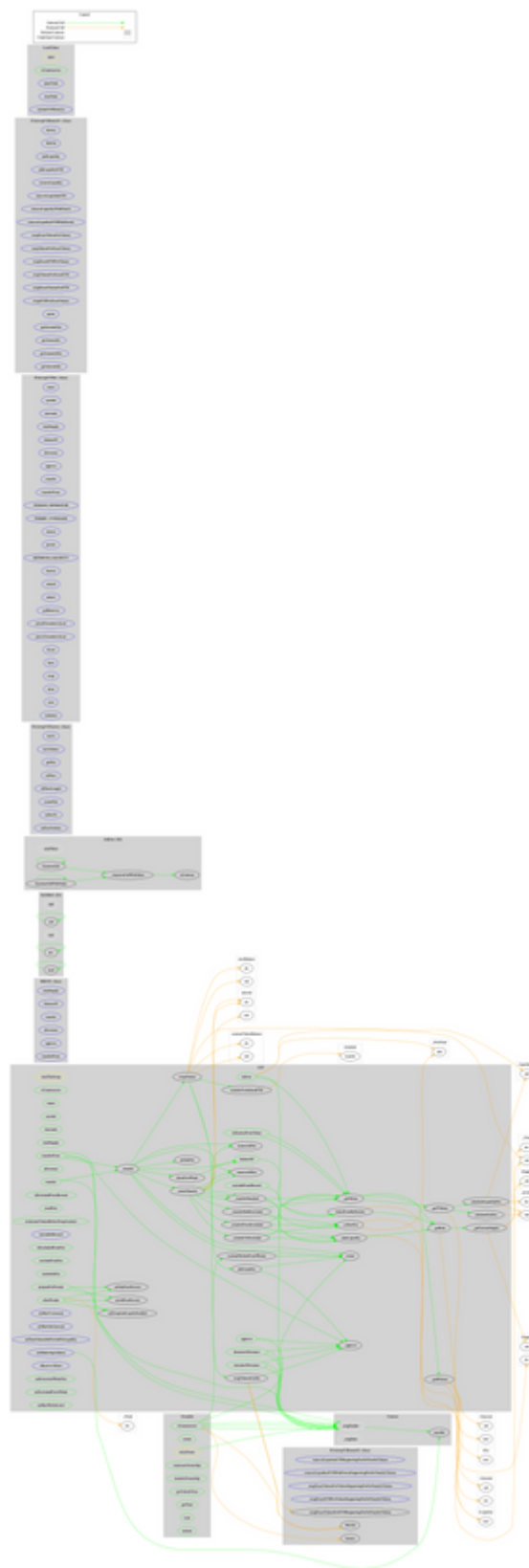
	nonces	External		-
	permit	External	✓	-
	MINIMUM_LIQUIDITY	External		-
	factory	External		-
	token0	External		-
	token1	External		-
	getReserves	External		-
	price0CumulativeLast	External		-
	price1CumulativeLast	External		-
	kLast	External		-
	burn	External	✓	-
	swap	External	✓	-
	skim	External	✓	-
	sync	External	✓	-
	initialize	External	✓	-
<b>IUniswapV2Router01</b>	Interface			
	factory	External		-
	WETH	External		-
	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	-
	quote	External		-
	getAmountOut	External		-
	getAmountIn	External		-
	getAmountsOut	External		-

	getAmountsIn	External		-
<b>IUniswapV2Router02</b>	Interface	IUniswapV2Router01		
	removeLiquidityETHSupportingFeeOnTransferTokens	External	✓	-
	removeLiquidityETHWithPermitSupportingFeeOnTransferTokens	External	✓	-
	swapExactTokensForTokensSupportingFeeOnTransferTokens	External	✓	-
	swapExactETHForTokensSupportingFeeOnTransferTokens	External	Payable	-
	swapExactTokensForETHSupportingFeeOnTransferTokens	External	✓	-
<b>LockToken</b>	Implementation	Ownable		
	<Constructor>	Public	✓	-
	openTrade	External	✓	onlyOwner
	stopTrade	External	✓	onlyOwner
	includeToWhiteList	External	✓	onlyOwner
<b>CST</b>	Implementation	Context, IBEP20, LockToken		
	<Constructor>	Public	✓	-
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	isExcludedFromReward	Public		-
	totalFees	Public		-

	minimumTokensBeforeSwapAmount	Public		-
	deliver	Public	✓	-
	reflectionFromToken	Public		-
	tokenFromReflection	Public		-
	excludeFromReward	Public	✓	onlyOwner
	includeInReward	External	✓	onlyOwner
	_approve	Private	✓	
	_transfer	Private	✓	open
	swapTokens	Private	✓	lockTheSwap
	swapTokensForEth	Private	✓	
	addLiquidity	Private	✓	
	_tokenTransfer	Private	✓	
	_transferStandard	Private	✓	
	_transferToExcluded	Private	✓	
	_transferFromExcluded	Private	✓	
	_transferBothExcluded	Private	✓	
	_reflectFee	Private	✓	
	_getValues	Private		
	_getTValues	Private		
	_getRValues	Private		
	_getRate	Private		
	_getCurrentSupply	Private		
	_takeLiquidity	Private	✓	
	calculateTaxFee	Private		
	calculateLiquidityFee	Private		
	isExcludedFromFee	Public		-
	excludeFromFee	Public	✓	onlyOwner
	includeInFee	Public	✓	onlyOwner
	removeAllFee	Private	✓	
	restoreAllFee	Private	✓	
	setSaleFee	Private	✓	
	setAllFeesPercent	Public	✓	onlyOwner
	setSaleFeesPercent	Public	✓	onlyOwner
	prepareForPresale	Public	✓	onlyOwner
	afterPresale	Public	✓	onlyOwner

	setMaxTxAmount	External	✓	onlyOwner
	setMaxSaleAmount	External	✓	onlyOwner
	setNumTokensSellToAddToLiquidity	External	✓	onlyOwner
	setMarketingAddress	External	✓	onlyOwner
	setSwapAndLiquifyEnabled	Public	✓	onlyOwner
	transferToAddressETH	Private	✓	
	<Receive Ether>	External	Payable	-
	excludeWalletsFromWhales	Private	✓	
	checkForWhale	Private	✓	
	setExtraAntiWhaleFee	Public	✓	onlyOwner
	setExcludedFromWhale	Public	✓	onlyOwner
	setMaxWalletLimit	Public	✓	onlyOwner

# Contract Flow



## Summary

The Smart Contract analysis of Community Support Token reported one high and one medium risk issue. There are some functions that can be abused by the owner, like manipulating fees and indirectly stopping the transactions. Additionally, there are some informative comments that do not affect the contract security. A multi-wallet signing pattern will provide security against potential hacks. Temporarily locking the contract or renouncing ownership will eliminate all the contract threats.

# 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>