Cyberscope

Audit Report

BC TOKEN

October 2022

Type	BEP20
Network	BSC
Address	0x5f17159F0e48115D7339c519411CF40857fd067A
Audited by	© cyberscope

Table of Contents

Contract Review

Contract Name	BCTOKEN
Compiler Version	v0.8.12+commit.f00d7308
Optimization	200 runs
Licence	None
Explorer	https://bscscan.com/token/0x5f17159f0e48115d7339c519411cf40857fd067a
Symbol	BCSC
Decimals	9
Total Supply	23,000,000
Domain	bcwallet.app

Source Files

Filename	SHA256
contract.sol	befd899e5a8e9bd6e8f88514edb5fb39c8da76ded836fe3dafbe0091fa05d57b

Audit Updates

Initial Audit	30th October 2022
Corrected	

Contract Analysis

● Critical ● Medium ● Minor / Informative ● Pass

Severity	Code	Description	Status
●	ST	Stops Transactions	Unresolved
●	OCTD	Transfers Contract's Tokens	Passed
●	OTUT	Transfers User's Tokens	Unresolved
●	ELFM	Exceeds Fees Limit	Passed
●	ULTW	Transfers Liquidity to Team Wallet	Passed
●	MT	Mints Tokens	Passed
●	BT	Burns Tokens	Passed
●	BC	Blacklists Addresses	Passed

ST - Stops Transactions

Criticality	minor / informative
Location	contract.sol#L355,374
Status	Unresolved

Description

The contract owner has the authority to stop the transactions for all users excluding the owner. The owner may take advantage of it by setting the `tradingOpen` to true without setting the `launchMode` to false.

```
if(!authorizations[sender] && !authorizations[recipient]){
    require(tradingOpen,"Trading not open yet");
}
```

The contract cal also stop the sales and transfers by setting the total fees to be equal to the burn fees.

Recommendation

The contract should not allow the contract owner to enable the tradingOpen variable after the initial toggle.

Read more about the Zero Division issue.

OTUT - Transfers User's Tokens

Criticality	critical
Location	contract.sol#L515
Status	Unresolved

Description

Any user has the authority to transfer the balance of a user’s contract to other addresses. The user may take advantage of it by calling the `multiTransfer` function before the `launchMode` toggle to false.

```
function multiTransfer(address from, address[] calldata addresses, uint256[]
calldata tokens) external authorized {
    if(msg.sender != from){
        require(launchMode,"Cannot execute this after launch is done");
    }

    require(addresses.length < 501,"GAS Error: max limit is 500 addresses");
    require(addresses.length == tokens.length,"Mismatch between address and
token count");

    uint256 SCCC = 0;

    for(uint i=0; i < addresses.length; i++){
        SCCC = SCCC + tokens[i];
    }

    require(balanceOf[from] >= SCCC, "Not enough tokens in wallet");

    for(uint i=0; i < addresses.length; i++){
        1(from,addresses[i],tokens[i]);
    }

}
```

Recommendation

The contract should not allow arbitrary tokens transfers.

Contract Diagnostics

🔴 Critical 🟠 Medium ● Minor / Informative

Severity	Code	Description	Status
🔴	ZD	Zero Division	Unresolved
●	L04	Conformance to Solidity Naming Conventions	Unresolved
●	L07	Missing Events Arithmetic	Unresolved
●	L13	Divide before Multiply Operation	Unresolved

ZD - Zero Division

Criticality	critical
Location	contract.sol#L374
Status	Unresolved

Description

The contract is using variables that may be set to zero as denominators. This may happen by setting the totalFee to zero or by setting the totalFee to equal the burnFee. As a result, the transactions will revert. This functionality is triggered during a sale or transfer transaction. Hence, the contract may prevent the users from selling.

```
uint256 totalETHFee = totalFee - burnFee;

uint256 amountToLiquify = (swapThreshold * liquidityFee)/(totalETHFee * 2);
```

Recommendation

The contract should prevent those variables to be set to zero or should not allow to execute the corresponding statements.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor / informative
Location	contract.sol#L483,162,148,175,497,540,118,256,463,444,436,471,549,262,109,548,156,550,546,428,355,552,541,539,506,452,363,551,544,158,159
Status	Unresolved

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.

```
_teamFeeReceiver
_allowances
WBNB
feeDenominator
_amount
Wallet_txExempt
WETH
maxWallPercent_base1000
_trans
...
```

Recommendation

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

L07 - Missing Events Arithmetic

Criticality	minor / informative
Location	contract.sol#L471,463
Status	Unresolved

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.

```
liquidityFee = _liquidityFee
sellMultiplier = _sell
```

Recommendation

Emit an event for critical parameter changes.

L13 - Divide before Multiply Operation

Criticality	minor / informative
Location	contract.sol#L300
Status	Unresolved

Description

Performing divisions before multiplications may cause lose of prediction.

```
feeAmount = amount.mul(totalFee).mul(multiplier).div(feeDenominator * 100)
```

Recommendation

The multiplications should be prior to the divisions.

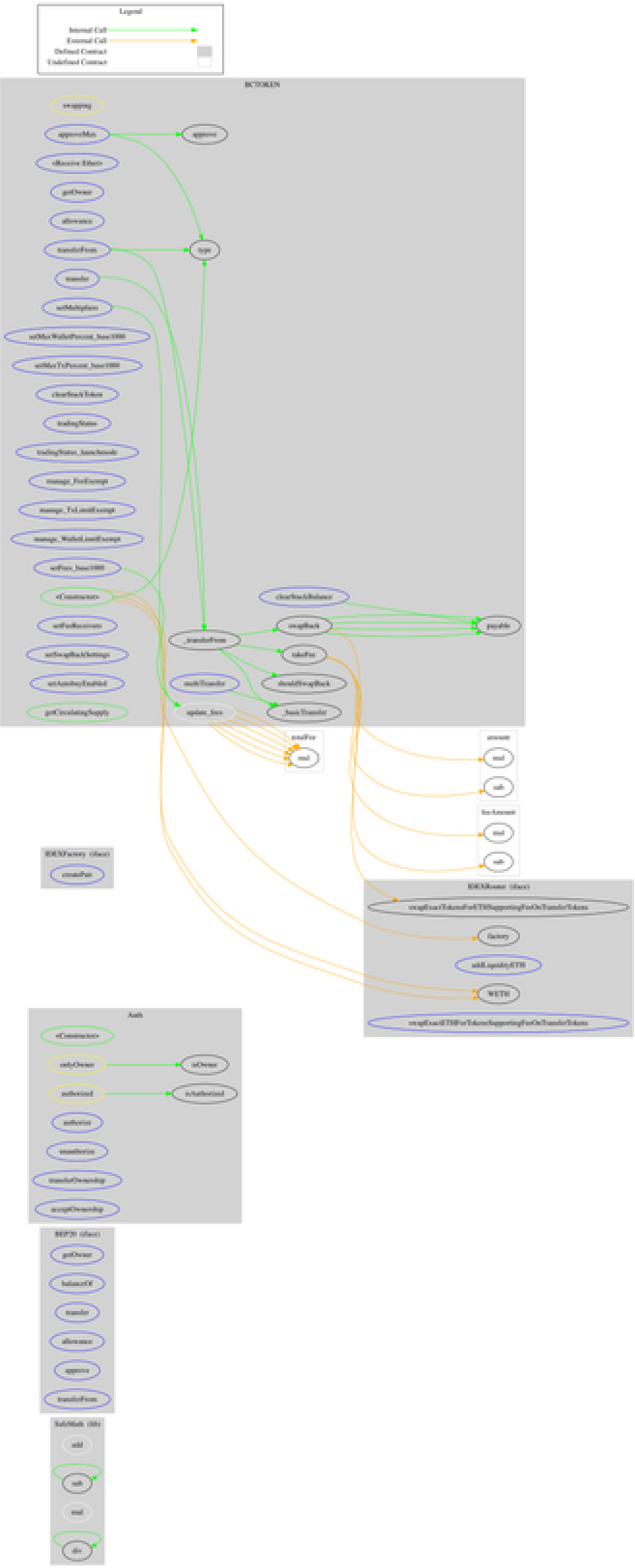
Contract Functions

Contract	**Type**	**Bases**		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
SafeMath	Library			
	add	Internal		
	sub	Internal		
	sub	Internal		
	mul	Internal		
	div	Internal		
	div	Internal		
BEP20	Interface			
	getOwner	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
Auth	Implementation			
	<Constructor>	Public	✓	-
	authorize	External	✓	onlyOwner
	unauthorize	External	✓	onlyOwner
	isOwner	Public		-
	isAuthorized	Public		-
	transferOwnership	External	✓	onlyOwner
	acceptOwnership	External	✓	-
IDEXFactory	Interface			
	createPair	External	✓	-
IDEXRouter	Interface			

	factory	External		-
	WETH	External		-
	addLiquidityETH	External	Payable	-
	swapExactTokensForETHSupportingFeeOnTransferTokens	External	✓	-
	swapExactETHForTokensSupportingFeeOnTransferTokens	External	Payable	-
BCTOKEN	Implementation	BEP20, Auth		
	<Constructor>	Public	✓	Auth
	<Receive Ether>	External	Payable	-
	getOwner	External		-
	allowance	External		-
	approve	Public	✓	-
	approveMax	External	✓	-
	transfer	External	✓	-
	transferFrom	External	✓	-
	setMaxWalletPercent_base1000	External	✓	onlyOwner
	setMaxTxPercent_base1000	External	✓	onlyOwner
	_transferFrom	Internal	✓	
	_basicTransfer	Internal	✓	
	takeFee	Internal	✓	
	shouldSwapBack	Internal		
	clearStuckBalance	External	✓	onlyOwner
	clearStuckToken	External	✓	onlyOwner
	tradingStatus	External	✓	onlyOwner
	tradingStatus_launchmode	External	✓	onlyOwner
	swapBack	Internal	✓	swapping
	manage_FeeExempt	External	✓	authorized
	manage_TxLimitExempt	External	✓	authorized
	manage_WalletLimitExempt	External	✓	authorized
	update_fees	Internal	✓	
	setMultipliers	External	✓	authorized
	setFees_base1000	External	✓	onlyOwner
	setFeeReceivers	External	✓	onlyOwner

	setSwapBackSettings	External	✓	onlyOwner
	setAutobuyEnabled	External	✓	onlyOwner
	getCirculatingSupply	Public		-
	multiTransfer	External	✓	authorized

Contract Flow



Domain Info

Domain Name	bcwallet.app
Registry Domain ID	49420F6AB-APP
Creation Date	2022-06-28T16:44:29Z
Updated Date	2022-07-04T18:46:51Z
Registry Expiry Date	2023-06-28T16:44:29Z
Registrar WHOIS Server	whois.godaddy.com
Registrar URL	https://www.godaddy.com/
Registrar	GoDaddy.com, LLC
Registrar IANA ID	146

The domain was created 4 months before the creation of the audit. It will expire in 8 months.

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

Summary

There are some functions that can be abused by the owner like stopping transactions and transferring the user's tokens. A multi-wallet signing pattern will provide security against potential hacks. Temporarily locking the contract or renouncing ownership will eliminate all the contract threats. There is also a limit of max 10% in buys, 20% on sales and 10% on transfer fees.

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.

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

About Cyberscope

Coinscope audit and K.Y.C. service has been rebranded to Cyberscope.

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.

Cyberscope and Coinscope are aiming to make crypto discoverable and efficient globally. They provide all the essential tools to assist users draw their own conclusions.

The Cyberscope team

https://www.cyberscope.io