



Cyberscope

Audit Report

Scamback

March 2022

Type BEP20

Network BSC

Address 0x140473341cAB53Ca9eC204A43155Fe420547A211

Audited by © cyberscope

Table of Contents

Table of Contents	1
Contract Review	3
Audit Updates	3
Contract Analysis	4
ST - Stop Transactions	5
Description	5
Recommendation	5
ELFM - Exceed Limit Fees Manipulation	7
Description	7
Recommendation	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

L13 - Divide before Multiply Operation	14
Description	14
Recommendation	14
Contract Functions	15
Contract Flow	21
Domain Info	22
Summary	23
Disclaimer	24
About Cyberscope	25

Contract Review

Contract Name	TOKEN
Compiler Version	v0.8.11+commit.d7f03943
Optimization	200 runs
Licence	Unlicense
Explorer	https://bscscan.com/token/0x140473341cAB53Ca9eC204A43155Fe420547A211
Symbol	\$BACK
Decimals	9
Total Supply	500,000,000,000
Source	contract.sol
Domain	scamback.com

Audit Updates

Initial Audit	3rd March 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

ST - Stop Transactions

Criticality	critical
Location	contract.sol#L1288

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 `_taxFee` to a high percentage value. If the sell Tax is increased over the allowed amount the contract may operate as a honeypot.

```
if (from == uniswapV2Pair) {  
    // Buy  
    _taxFee = _buyTaxFee;  
    _liquidityFee = _buyLiquidityFee;  
    _marketingFee = _buyMarketingFee;  
    _charityFee = _buyChairtyFee;  
} else if (to == uniswapV2Pair) {  
    // Sell  
    _taxFee = _sellTaxFee;  
    _liquidityFee = _sellLiquidityFee;  
    _marketingFee = _sellMarketingFee;  
    _charityFee = _sellCharityFee;  
} else {  
    // Transfer  
    _taxFee = _buyTaxFee;  
    _liquidityFee = _buyLiquidityFee;  
    _marketingFee = _buyMarketingFee;  
    _charityFee = _buyChairtyFee;  
}  
}
```

Recommendation

The contract could embody a check for not allowing setting the `_taxFee` more 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

Criticality	critical
Location	contract.sol#L1003,1015

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 `setSellFeePercent` function with a high percentage value.

```
function setSellFeePercent(
    uint256 tFee,
    uint256 lFee,
    uint256 mFee,
    uint256 cFee
) external onlyOwner {
    _sellTaxFee = tFee;
    _sellLiquidityFee = lFee;
    _sellMarketingFee = mFee;
    _sellCharityFee = cFee;
}
```

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
●	L04	Conformance to Solidity Naming Conventions
●	L09	Dead Code Elimination
●	L07	Missing Events Arithmetic
●	L13	Divide before Multiply Operation

L01 - Public Function could be Declared External

Criticality	minor
Location	contract.sol#L318,323,333,341,801,805,809,813,822,831 and 12 more

Description

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

```
isExcludedFromFee  
includeInFee  
excludeFromFee  
excludeFromReward  
reflectionFromToken  
deliver  
totalFees  
isExcludedFromReward  
decreaseAllowance  
...
```

Recommendation

Use the external attribute for functions never called from the contract

L02 - State Variables could be Declared Constant

Criticality	minor
Location	contract.sol#L709

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#L295,297,415,417,448,494,1021,1025,1043,1192 and 9 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.

```
_charityFee  
_marketingFee  
_liquidityFee  
_taxFee  
_burnAddress  
_charityWalletAddress  
_marketingWalletAddress  
_amount  
_enabled  
...
```

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#L274,183,190,198,212,252,264,229,242,163 and 10 more

Description

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

```
trySub  
tryMul  
tryMod  
tryDiv  
tryAdd  
mod  
div  
_msgData  
sendValue  
...
```

Recommendation

Remove unused functions.

L07 - Missing Events Arithmetic

Criticality

minor

Location

contract.sol#L997,1009,1029

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.

```
numTokensSellToAddToLiquidity = amount * 10 ** _decimals
_buyTaxFee = tFee
_sellTaxFee = tFee
```

Recommendation

Emit an event for critical parameter changes.

L13 - Divide before Multiply Operation

Criticality

minor

Location

contract.sol#L1298

Description

Performing divisions before multiplications may cause lose of prediction.

```
halfFunds = newBalance.div(_totalFees).mul(_liquidityFee.div(2))  
chairtyFunds = newBalance.div(_totalFees).mul(_charityFee)  
marketingFunds = newBalance.div(_totalFees).mul(_marketingFee)  
charityTokens = contractTokenBalance.div(_totalFees).mul(_charityFee)  
marketingTokens = contractTokenBalance.div(_totalFees).mul(_marketingFee)
```

Recommendation

The multiplications should be prior to the divisions.

Contract Functions

Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
SafeMath	Library			
	tryAdd	Internal		
	trySub	Internal		
	tryMul	Internal		
	tryDiv	Internal		
	tryMod	Internal		
	add	Internal		
	sub	Internal		
	mul	Internal		
	div	Internal		
	mod	Internal		
	sub	Internal		
	div	Internal		
	mod	Internal		
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
Address	Library			
	isContract	Internal		

	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionStaticCall	Internal		
	functionStaticCall	Internal		
	functionDelegateCall	Internal	✓	
	functionDelegateCall	Internal	✓	
	_verifyCallResult	Private		
Ownable	Implementation	Context		
	<Constructor>	Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
	lock	Public	✓	onlyOwner
	unlock	Public	✓	-
IUniswapV2Factory	Interface			
	feeTo	External		-
	feeToSetter	External		-
	getPair	External		-
	allPairs	External		-
	allPairsLength	External		-
	createPair	External	✓	-
	setFeeTo	External	✓	-
	setFeeToSetter	External	✓	-
IUniswapV2Pair	Interface			
	name	External		-
	symbol	External		-
	decimals	External		-
	totalSupply	External		-

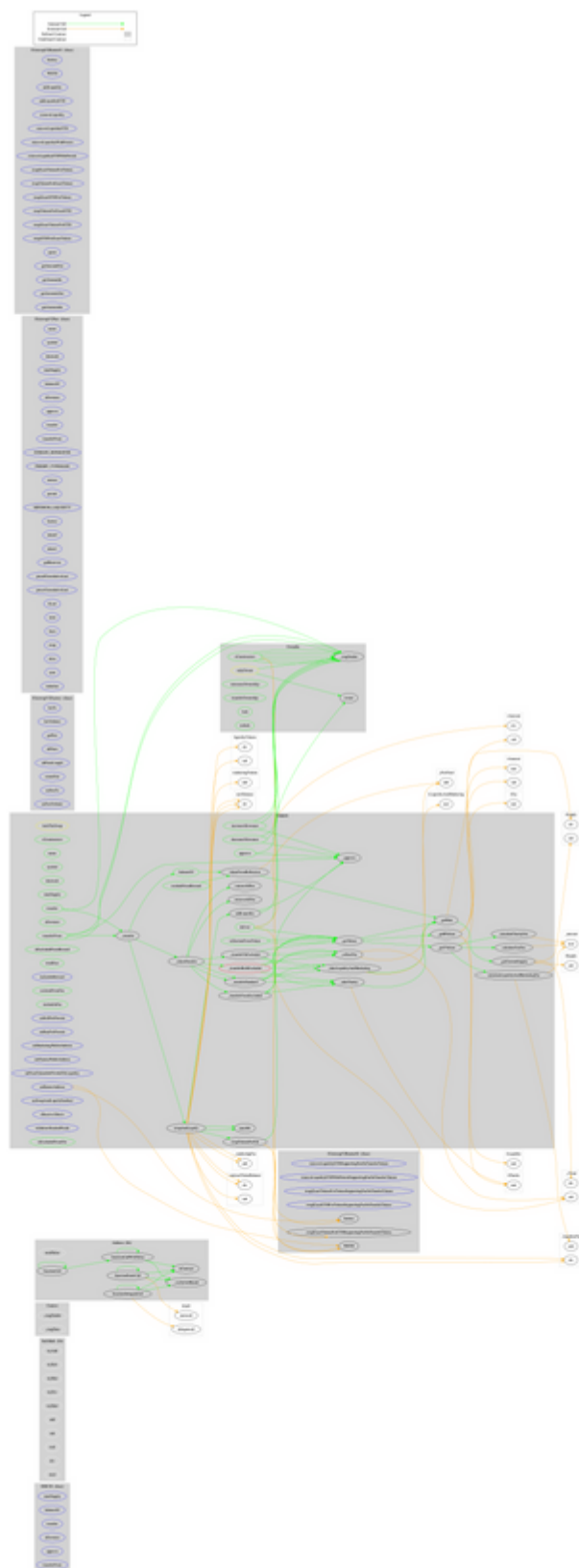
	balanceOf	External		-
	allowance	External		-
	approve	External	✓	-
	transfer	External	✓	-
	transferFrom	External	✓	-
	DOMAIN_SEPARATOR	External		-
	PERMIT_TYPEHASH	External		-
	nonces	External		-
	permit	External	✓	-
	MINIMUM_LIQUIDITY	External		-
	factory	External		-
	token0	External		-
	token1	External		-
	getReserves	External		-
	price0CumulativeLast	External		-
	price1CumulativeLast	External		-
	kLast	External		-
	mint	External	✓	-
	burn	External	✓	-
	swap	External	✓	-
	skim	External	✓	-
	sync	External	✓	-
	initialize	External	✓	-
IUniswapV2Router01	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		-
IUniswapV2Router02	Interface	IUniswapV2Router01		
	removeLiquidityETHSupportingFeeOnTransferTokens	External	✓	-
	removeLiquidityETHWithPermitSupportingFeeOnTransferTokens	External	✓	-
	swapExactTokensForTokensSupportingFeeOnTransferTokens	External	✓	-
	swapExactETHForTokensSupportingFeeOnTransferTokens	External	Payable	-
	swapExactTokensForETHSupportingFeeOnTransferTokens	External	✓	-
TOKEN	Implementation	Context, IERC20, Ownable		
	<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		-
	deliver	Public	✓	-
	reflectionFromToken	Public		-
	tokenFromReflection	Public		-
	excludeFromReward	Public	✓	onlyOwner
	includeInReward	External	✓	onlyOwner
	_transferBothExcluded	Private	✓	
	excludeFromFee	Public	✓	onlyOwner
	includeInFee	Public	✓	onlyOwner
	setSellFeePercent	External	✓	onlyOwner
	setBuyFeePercent	External	✓	onlyOwner
	setMarketingWalletAddress	External	✓	onlyOwner
	setCharityWalletAddress	External	✓	onlyOwner
	setNumTokensSellToAddToLiquidity	External	✓	onlyOwner
	setRouterAddress	External	✓	onlyOwner
	setSwapAndLiquifyEnabled	External	✓	onlyOwner
	<Receive Ether>	External	Payable	-
	withdrawStuckedFunds	External	✓	onlyOwner
	_reflectFee	Private	✓	
	_getValues	Private		
	_getTValues	Private		
	_getRValues	Private		
	_getRate	Private		
	_getCurrentSupply	Private		
	_takeLiquidityAndMarketing	Private	✓	
	_takeCharity	Private	✓	
	calculateTaxFee	Private		
	calculateCharityFee	Private		
	calculateLiquidityAndMarketingFee	Private		
	removeAllFee	Private	✓	
	restoreAllFee	Private	✓	
	isExcludedFromFee	Public		-
	_approve	Private	✓	
	_transfer	Private	✓	

	swapAndLiquify	Private	✓	lockTheSwap
	swapTokensForEth	Private	✓	
	addLiquidity	Private	✓	
	_tokenTransfer	Private	✓	
	_transferStandard	Private	✓	
	_transferToExcluded	Private	✓	
	_transferFromExcluded	Private	✓	

Contract Flow



Domain Info

Domain Name	scamback.com
Registry Domain ID	2674575640_DOMAIN_COM-VRSN
Creation Date	2022-02-11T20:52:14.000Z
Updated Date	2022-02-11T20:52:16.000Z
Registry Expiry Date	
Registrar WHOIS Server	whois.ionos.com
Registrar URL	http://ionos.com
Registrar	IONOS SE
Registrar IANA ID	83

The domain has been created 20 days before the creation of the audit.

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 manipulating fees and stopping transactions. If the contract configuration is abused by the owner, the contract may operate as a honeypot. 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.

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 provides all the essential tools to assist users draw their own conclusions.



The Cyberscope team

<https://www.cyberscope.io>