Cyberscope

Audit Report

SoccerCat

July 2022

Type	BEP20
Network	BSC
Address	0xa056269e5ab784754ee33dad0051e056e662fd6d
Audited by	© cyberscope

Table of Contents

Contract Review

Contract Name	SoccerCat
Compiler Version	v0.8.7+commit.e28d00a7
Optimization	200 runs
Licence	None
Explorer	https://bscscan.com/token/0xa056269e5ab784754ee33dad0051e056e662fd6d
Symbol	SCAT
Decimals	18
Total Supply	10,000,000,000
Domain	soccercat.app

Source Files

Filename	**SHA256**
contract.sol	a731139d038461040675a1d3a38a0deb996bc73f4c88d3923ae1db3aab697b3c

Audit Updates

Initial Audit	20th July 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

Contract Diagnostics

Critical Medium Minor

Severity	Code	Description
	STC	Succeeded Transfer Check
	CR	Code Repetition
	L01	Public Function could be Declared External
	L02	State Variables could be Declared Constant
	L04	Conformance to Solidity Naming Conventions
	L07	Missing Events Arithmetic
	L09	Dead Code Elimination

STC - Succeeded Transfer Check

Criticality	minor
Location	contract.sol#L701

Description

According to the ERC20 specification, the transfer methods should be checked if the result is successful. Otherwise, the contract may wrongly assume that the transfer has been established.

```
function transferToAddressETH(address payable recipient, uint256 amount) private {
        recipient.transfer(amount);
}
```

Recommendation

The contract should check if the result of the transfer methods is successful.

CR - Code Repetition

Criticality	minor
Location	contract.sol#L850

Description

There are code segments that are repetitive in the contract. Those segments increase the code size of the contract unnecessarily.

```
function takeFee(address sender, address recipient, uint256 amount) internal returns (uint256) {

    uint256 feeAmount = 0;
    uint256 destAmount = 0;

    if(isMarketPair[sender]) {
        feeAmount = amount.mul(_totalTaxIfBuying.sub(_buyDestroyFee)).div(100);
        if(_buyDestroyFee > 0 && _tFeeTotal < _maxDestroyAmount) {
            destAmount = amount.mul(_buyDestroyFee).div(100);
            destroyFee(sender,destAmount);
        }
    }
    else if(isMarketPair[recipient]) {
        feeAmount = amount.mul(_totalTaxIfSelling.sub(_sellDestroyFee)).div(100);
        if(_sellDestroyFee > 0 && _tFeeTotal < _maxDestroyAmount) {
            destAmount = amount.mul(_sellDestroyFee).div(100);
            destroyFee(sender,destAmount);
        }
    }

    if(feeAmount > 0) {
        _balances[address(this)] = _balances[address(this)].add(feeAmount);
        emit Transfer(sender, address(this), feeAmount);
    }

    return amount.sub(feeAmount.add(destAmount));
}
```

Recommendation

Create an internal function that contains the code segment and remove it from all the sections.

L01 - Public Function could be Declared External

Criticality	minor
Location	contract.sol#L684,579,689,575,559,166,618,563,584,723,177,171,701,555,593,589,728,606,627,614,693,622,567

Description

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

```
totalSupply
setBuyDestFee
getCirculatingSupply
setIsExcludedFromFee
setSellDestFee
setMarketPairStatus
transferFrom
minimumTokensBeforeSwapAmount
approve
...
```

Recommendation

Use the external attribute for functions never called from the contract.

L02 - State Variables could be Declared Constant

Criticality	minor
Location	contract.sol#L435

Description

Constant state variables should be declared constant to save gas.

```
deadAddress
```

Recommendation

Add the constant attribute to state variables that never change.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contract.sol#L461,453,452,466,227,456,467,244,446,457,447,225,450,464,152,437,684,460,463,451,448,458,274,445,455

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.

```
_liquidityShare
_buyLiquidityFee
WETH
_totalDistributionShares
_buyDestroyFee
_sellMarketingFee
_tFeeTotal
_totalTaxIfBuying
_enabled
...
```

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
Location	contract.sol#L632,672,622,656,618,668,627,640,648

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.

```
_liquidityShare = newLiquidityShare
_totalTaxIfSelling =
_sellLiquidityFee.add(_sellMarketingFee).add(_sellTeamFee).add(_sellDestroyFee)
_sellDestroyFee = newSellDestroyFee
_walletMax = newLimit
_maxDestroyAmount = maxDestroy
_maxTxAmount = maxTxAmount
_buyDestroyFee = newBuyDestroyFee
_minimumTokensBeforeSwap = newLimit
_totalTaxIfBuying =
_buyLiquidityFee.add(_buyMarketingFee).add(_buyTeamFee).add(_buyDestroyFee)
```

Recommendation

Emit an event for critical parameter changes.

L09 - Dead Code Elimination

Criticality	minor
Location	contract.sol#L106,122,126,131,95,118,114

Description

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

```
functionCall
isContract
_functionCallWithValue
functionCallWithValue
sendValue
```

Recommendation

Remove unused functions.

Contract Functions

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

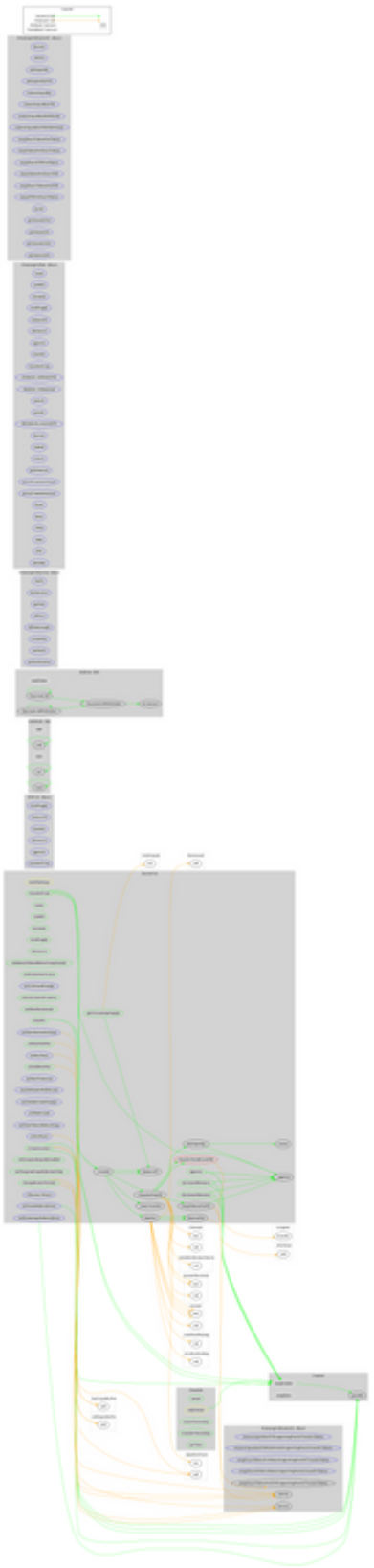
	_functionCallWithValue	Private	✓	
Ownable	Implementation	Context		
	owner	Public		-
	waiveOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
	getTime	Public		-
IUniswapV2Factory	Interface			
	feeTo	External		-
	feeToSetter	External		-
	getPair	External		-
	allPairs	External		-
	allPairsLength	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		-
	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	✓	-
SoccerCat	Implementation	Context, IERC20, Ownable		
	<Constructor>	Public	Payable	-
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	allowance	Public		-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	minimumTokensBeforeSwapAmount	Public		-
	approve	Public	✓	-
	_approve	Private	✓	
	setMarketPairStatus	Public	✓	onlyOwner
	setIsTxLimitExempt	External	✓	onlyOwner
	setIsExcludedFromFee	Public	✓	onlyOwner
	setMaxDesAmount	Public	✓	onlyOwner
	setBuyDestFee	Public	✓	onlyOwner
	setSellDestFee	Public	✓	onlyOwner
	setBuyTaxes	External	✓	onlyOwner
	setSelTaxes	External	✓	onlyOwner
	setDistributionSettings	External	✓	onlyOwner
	setMaxTxAmount	External	✓	onlyOwner
	enableDisableWalletLimit	External	✓	onlyOwner
	setIsWalletLimitExempt	External	✓	onlyOwner

	setWalletLimit	External	✓	onlyOwner
	setNumTokensBeforeSwap	External	✓	onlyOwner
	setMarketingWalletAddress	External	✓	onlyOwner
	setTeamWalletAddress	External	✓	onlyOwner
	setSwapAndLiquifyEnabled	Public	✓	onlyOwner
	setSwapAndLiquifyByLimitOnly	Public	✓	onlyOwner
	getCirculatingSupply	Public		-
	transferToAddressETH	Private	✓	
	changeRouterVersion	Public	✓	onlyOwner
	<Receive Ether>	External	Payable	-
	transfer	Public	✓	-
	transferFrom	Public	✓	-
	_transfer	Private	✓	
	_basicTransfer	Internal	✓	
	swapAndLiquify	Private	✓	lockTheSwap
	swapTokensForEth	Private	✓	
	addLiquidity	Private	✓	
	takeFee	Internal	✓	
	destroyFee	Private	✓	

Contract Flow



Domain Info

Domain Name	soccercat.app
Registry Domain ID	499B4FEB9-APP
Creation Date	2022-07-19T13:46:46Z
Updated Date	2022-07-19T14:21:23Z
Registry Expiry Date	2023-07-19T13:46:46Z
Registrar WHOIS Server	whois.nic.google
Registrar URL	http://www.tucows.com
Registrar	Tucows Domains Inc
Registrar IANA ID	69

The domain has been created in 12 months before the creation of the audit.

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

Summary

SoccerCat Token is an interesting project that has a friendly and growing community. The Smart Contract analysis reported no compiler error or critical issues. The contract Owner can not access admin functions. The Ownership of the contact is renounced.

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