Cyberscope

Audit Report

Champion

July 2022

Type	BEP20
Network	BSC
Address	0xFd159d6bE08128D5a3776286e14c49703133B47f
Audited by	© cyberscope

Table of Contents

Contract Review

Contract Name	CH
Compiler Version	v0.8.15+commit.e14f2714
Optimization	200 runs
Licence	MIT
Explorer	https://bscscan.com/token/0xFd159d6bE08128D5a3776286e14c49703133B47f
Symbol	CH
Decimals	18
Total Supply	30,000,000
Domain	championcoin.club

Source Files

Filename	**SHA256**
contract.sol	a51f04951793645c3f20791d2ebe4c1f37a86a17f8c756b8dc78ff0b686e91a7

Audit Updates

Initial Audit	8th 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

ST - Stop Transactions

Criticality	critical
Location	contract.sol#L302,323

Description

The contract owner has the authority to stop the sales for all users excluding the owner. The owner may take advantage of it by setting the variables that sum up the `swapFee` to zero. As a result, the swapFee will produce a zero division revert.

```
uint256 swapFee = _sellLPFee + _sellMarketingFee + _buyLPFee + _buyMarketingFee;
uint256 lpFee = _buyLPFee + _sellLPFee;
uint256 lpAmount = tokenAmount * lpFee / swapFee;
```

The contract owner may also take advantage of it by increasing the _sellLPFee to a high value. As a result, the sender's amount will not be sufficient and the transaction will revert.

```
if (isSell) {
    swapFee = _sellLPFee + _sellMarketingFee;
} else {
    swapFee = _buyLPFee + _buyMarketingFee;
}
uint256 swapAmount = tAmount * swapFee / 100;
```

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.

Regarding the fees read more in the fees manipulation section.

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#L383

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 `setSellLPFee` function with a high percentage value.

```
function setSellLPFee(uint256 lpFee) external onlyOwner {
    _sellLPFee = lpFee;
}
```

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
	STC	Succeeded Transfer Check
	L01	Public Function could be Declared External
	L02	State Variables could be Declared Constant
	L04	Conformance to Solidity Naming Conventions
	L05	Unused State Variable
	L06	Missing Events Access Control
	L14	Uninitialized Variables in Local Scope

STC - Succeeded Transfer Check

Criticality	minor
Location	contract.sol#L339

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.

```
if(marketingAmount>0){

BUSD.transferFrom(address(_tokenDistributor),marketingAddress,marketingAmount);
}
uint256 amountBUSDLiquidity = BUSDBalance - marketingAmount;
BUSD.transferFrom(address(_tokenDistributor), address(this),
amountBUSDLiquidity);
```

Recommendation

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

L01 - Public Function could be Declared External

Criticality	minor
Location	contract.sol#L76,85,90,197,205,210,214,219

Description

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

```
transferFrom
approve
allowance
transfer
totalSupply
transferOwnership
renounceOwnership
owner
```

Recommendation

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

L02 - State Variables could be Declared Constant

Criticality	minor
Location	contract.sol#L126,125,133

Description

Constant state variables should be declared constant to save gas.

```
sharetotal
router
BUSDaddress
```

Recommendation

Add the constant attribute to state variables that never change.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contract.sol#L39,66,111,118,120,121,122,123,126,127,128,129,130

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.

```
_swapPairList
_mainPair
_swapRouter
BUSD
BUSDaddress
_sellMarketingFee
_sellLPFee
_buyMarketingFee
_buyLPFee
...
```

Recommendation

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

L05 - Unused State Variable

Criticality	minor
Location	contract.sol#L133

Description

There are segments that contain unused state variables.

```
sharetotal
```

Recommendation

Remove unused state variables.

L06 - Missing Events Access Control

Criticality	minor
Location	contract.sol#L362

Description

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

```
fundAddress = addr
```

Recommendation

Emit an event for critical parameter changes.

L14 - Uninitialized Variables in Local Scope

Criticality	minor
Location	contract.sol#L242,241,297

Description

The are variables that are defined in the local scope and are not initialized.

```
feeAmount
isSell
takeFee
```

Recommendation

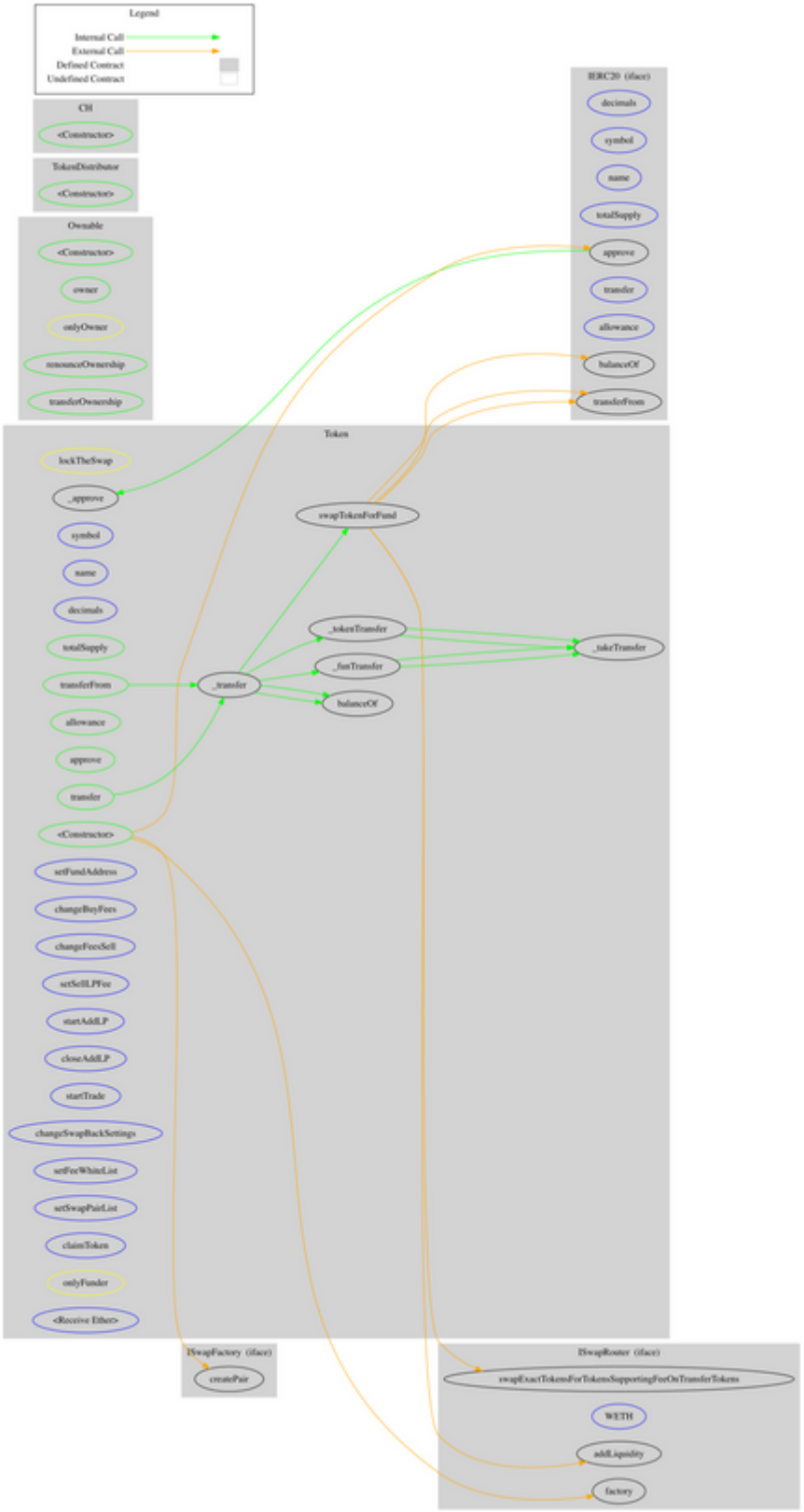
All the local scoped variables should be initialized.

Contract Functions

Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
IERC20	Interface			
	decimals	External		-
	symbol	External		-
	name	External		-
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
ISwapRouter	Interface			
	factory	External		-
	WETH	External		-
	swapExactTokensForTokensSupportingFeeOnTransferTokens	External	✓	-
	addLiquidity	External	✓	-
ISwapFactory	Interface			
	createPair	External	✓	-
Ownable	Implementation			
	<Constructor>	Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
TokenDistributor	Implementation			

	<Constructor>	Public	✓	-
Token	Implementation	IERC20, Ownable		
	<Constructor>	Public	✓	-
	symbol	External		-
	name	External		-
	decimals	External		-
	totalSupply	Public		-
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	_approve	Private	✓	
	_transfer	Private	✓	
	_funTransfer	Private	✓	
	_tokenTransfer	Private	✓	
	swapTokenForFund	Private	✓	lockTheSwap
	_takeTransfer	Private	✓	
	setFundAddress	External	✓	onlyFunder
	changeBuyFees	External	✓	onlyOwner
	changeFeesSell	External	✓	onlyOwner
	setSellLPFee	External	✓	onlyOwner
	startAddLP	External	✓	onlyOwner
	closeAddLP	External	✓	onlyOwner
	startTrade	External	✓	onlyOwner
	changeSwapBackSettings	External	✓	onlyOwner
	setFeeWhiteList	External	✓	onlyFunder
	setSwapPairList	External	✓	onlyFunder
	claimToken	External	✓	onlyFunder
	<Receive Ether>	External	Payable	-
CH	Implementation	Token		
	<Constructor>	Public	✓	Token

Contract Flow



Domain Info

Domain Name	championcoin.club
Registry Domain ID	D6C647AB0DE474FF082AFADCF2B243596-GDREG
Creation Date	2022-06-23T13:16:24Z
Updated Date	2022-06-28T13:16:25Z
Registry Expiry Date	2023-06-23T13:16:24Z
Registrar WHOIS Server	whois.dnspod.cn
Registrar URL	www.dnspod.cn
Registrar	DNSPod, Inc.
Registrar IANA ID	1697

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

There are some functions that can be abused by the owner like stopping transactions and manipulating fees. The contract can be converted into a honeypot and prevent users from selling if the owner abuses the admin functions. 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 provide all the essential tools to assist users draw their own conclusions.

The Cyberscope team

https://www.cyberscope.io