



Cyberscope

Audit Report

Liver Bird

April 2022

Type BEP20

Network BSC

Address 0x33E509F0b9a3A863f06B4124EF8BfdBCEC161D0C

Audited by © cyberscope

Table of Contents

Table of Contents	1
Contract Review	3
Source Files	3
Audit Updates	3
Contract Analysis	4
OCTD - Owner Contract Tokens Drain	5
Description	5
Recommendation	5
Resolved	5
Contract Diagnostics	6
L01 - Public Function could be Declared External	7
Description	7
Recommendation	7
L02 - State Variables could be Declared Constant	8
Description	8
Recommendation	8
L04 - Conformance to Solidity Naming Conventions	9
Description	9
Recommendation	9
L05 - Unused State Variable	10
Description	10
Recommendation	10
Contract Functions	11
Contract Flow	14
Domain Info	15
Summary	16

Disclaimer**17****About Cyberscope****18**

Contract Review

Contract Name	LiverBird
Compiler Version	v0.8.13+commit.abaa5c0e
Optimization	200 runs
Licence	Unlicense
Explorer	https://bscscan.com/token/0x33E509F0b9a3A863f06B4124EF8BfdBCEC161D0C
Symbol	LiverBi
Decimals	9
Total Supply	1,000,000,000
Domain	liverbird.club

Source Files

Filename	SHA256
contract.sol	185e2efd6f328ab5037f305609009fa47e0c41a09e9a8a06699cf679d8a13914

Audit Updates

Initial Audit	20th April 2022
Corrected	

Contract Analysis

● Critical ● Medium ● Minor ● Pass

Severity	Code	Description	Status
●	ST	Contract Owner is not able to stop or pause transactions	
●	OCTD	Contract Owner is not able to transfer tokens from specific address	Resolved
●	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	

OCTD - Owner Contract Tokens Drain

Criticality	minor
Location	contract.sol#L352
Status	Resolved

Description

The contract owner has the authority to claim all the balance of the contract. The owner may take advantage of it by calling the `clearBNBStuck` function.

```
function clearBNBStuck() external onlyDev {  
    uint256 balance = address(this).balance;  
    payable(_marketingAddress).transfer(balance);  
}
```

Recommendation

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.

Resolved

The team has renounced ownership and resolved the threat.

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
●	L05	Unused State Variable

L01 - Public Function could be Declared External

Criticality

minor

Location

contract.sol#L106,112,167,171,175,179,187,192,196,201,275,283,337,347

Description

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

```
excludeMultipleAccountsFromFees  
setFee  
setNewDevAddress  
setNewMarketingAddress  
transferFrom  
approve  
allowance  
transfer  
totalSupply  
...
```

Recommendation

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

L02 - State Variables could be Declared Constant

Criticality

minor

Location

contract.sol#L129,89

Description

Constant state variables should be declared constant to save gas.

```
_previousOwner  
_rTotal
```

Recommendation

Add the constant attribute to state variables that never change.

L04 - Conformance to Solidity Naming Conventions

Criticality

minor

Location

contract.sol#L30,273,281,128,137,139,140,141

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.

```
_decimals  
_symbol  
_name  
_FeeTotal  
_tTotal  
devAddressUpdated  
marketingAddressUpdated  
WETH
```

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#L89,123

Description

There are segments that contain unused state variables.

```
_tOwned  
_previousOwner
```

Recommendation

Remove unused state variables.

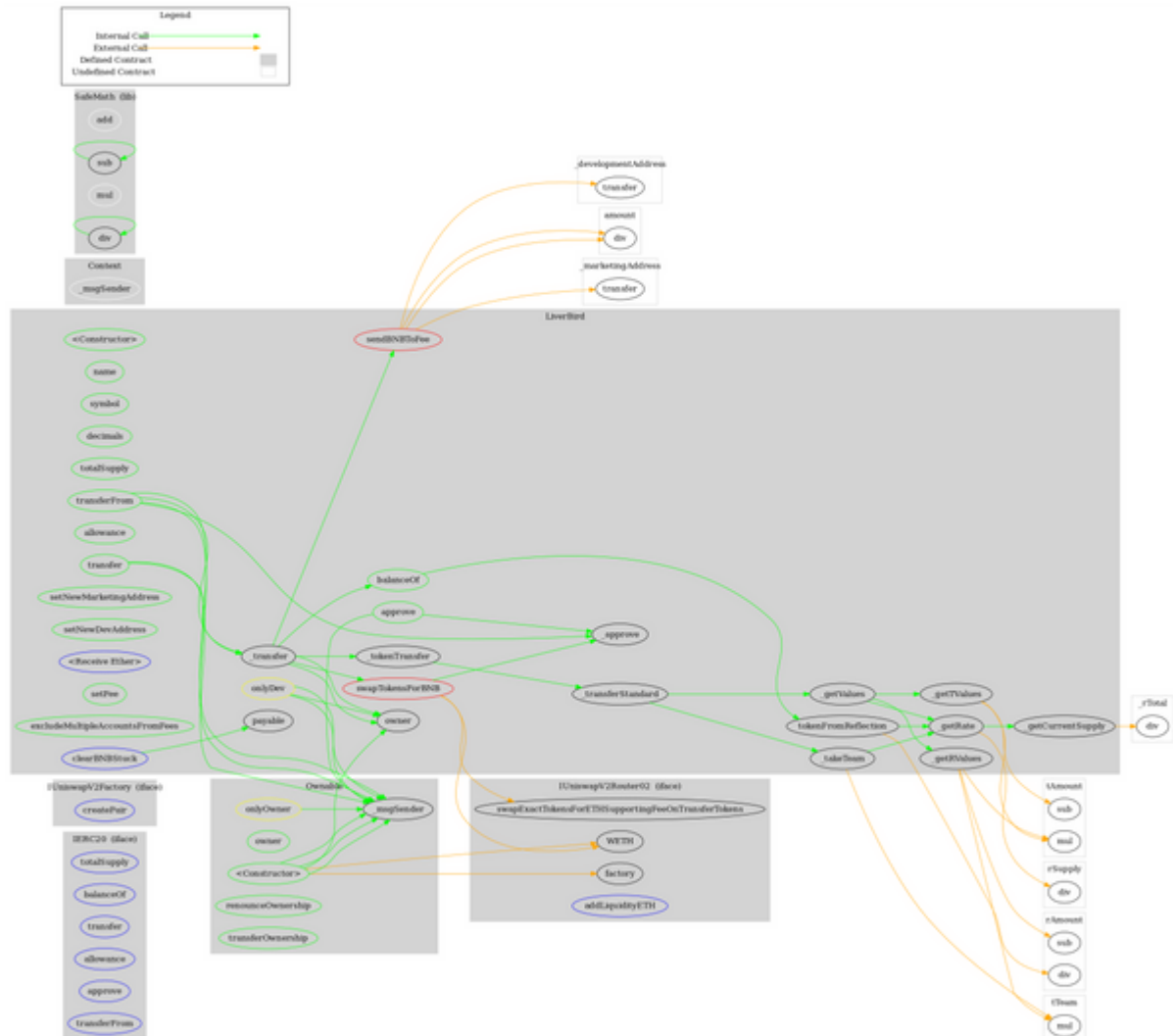
Contract Functions

Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
IUniswapV2Factory	Interface			
	createPair	External	✓	-
IUniswapV2Router02	Interface			
	swapExactTokensForETHSupportingFeeOnTransferTokens	External	✓	-
	factory	External		-
	WETH	External		-
	addLiquidityETH	External	Payable	-
Context	Implementation			
	_msgSender	Internal		
SafeMath	Library			
	add	Internal		
	sub	Internal		
	sub	Internal		
	mul	Internal		
	div	Internal		
	div	Internal		

Ownable	Implementation	Context		
	<Constructor>	Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
LiverBird	Implementation	Context, IERC20, Ownable		
	<Constructor>	Public	✓	-
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	tokenFromReflection	Private		
	_approve	Private	✓	
	swapTokensForBNB	Private	✓	
	sendBNBToFee	Private	✓	
	_transfer	Private	✓	
	_tokenTransfer	Private	✓	
	setNewMarketingAddress	Public	✓	onlyDev
	setNewDevAddress	Public	✓	onlyDev
	_transferStandard	Private	✓	
	_takeTeam	Private	✓	
	<Receive Ether>	External	Payable	-
	_getValues	Private		
	_getTValues	Private		
	_getRValues	Private		
	_getRate	Private		
	_getCurrentSupply	Private		

	setFee	Public	✓	onlyDev
	excludeMultipleAccountsFromFees	Public	✓	onlyDev
	clearBNBStuck	External	✓	onlyDev

Contract Flow



Domain Info

Domain Name	liverbird.club
Registry Domain ID	DC4291FD0B1CF4A299C5F048F3AECE61-GDREG
Creation Date	2022-04-18T14:22:05Z
Updated Date	2022-04-18T14:23:57Z
Registry Expiry Date	2023-04-18T14:22:05Z
Registrar WHOIS Server	whois.namecheap.com
Registrar URL	http://www.namecheap.com
Registrar	NameCheap, Inc.
Registrar IANA ID	1068

The domain has been created 2 days before the creation of the audit. It will expire in 12 months.

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

Summary

Liver Bird is an interesting project that has a friendly and growing community. The Smart Contract analysis reported no compiler error or critical issues. The team has renounced ownership and can no longer interact with the contract functions. The fees are fixed to 5% and can't be changed.

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>