



Audit Report

Wenabis

January 2022

Type	BEP20
Network	BSC
Address	0x011298d0F715c1571f6947EF40Ebb3Afd4AC4E3F
Audited by	© coinscope

Table of Contents

Table of Contents	1
Contract Review	2
Audit Updates	2
Contract Analysis	3
ELFM - Exceed Limit Fees Manipulation	4
Description	4
Recommendation	4
LTW - Liquidity to Team Wallet	4
Description	5
Recommendation	5
Contract Diagnostics	6
Contract Functions	7
Contract Flow	20
Domain Info	21
Summary	22
Disclaimer	23
About Coinscope	24

Contract Review

Contract Name	Wenabisv2
Compiler Version	v0.8.9+commit.e5eed63a
Optimization	200 runs
Licence	
Explorer	https://bscscan.com/token/0x011298d0F715c1571f6947EF40Ebb3Afd4AC4E3F
Symbol	WNB
Decimals	18
Total Supply	69,981,416
Source	contracts/Wenabisv2.sol, @openzeppelin/contracts/token/ERC20/ERC20.sol, @openzeppelin/contracts/token/ERC20/IERC20.sol, @openzeppelin/contracts/access/Ownable.sol, @openzeppelin/contracts/security/Pausable.sol, @openzeppelin/contracts/token/ERC20/extensions/ERC20Burnable.sol, @openzeppelin/contracts/utils/Context.sol, @openzeppelin/contracts/utils/Address.sol, @openzeppelin/contracts/utils/math/SafeMath.sol, @openzeppelin/contracts/token/ERC20/extensions/IERC20Metadata.sol
Domain	wenabis.co

Audit Updates

Initial Audit	25th January 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

MT - Mint Tokens

Criticality	critical
Location	contract.sol#L1,L892

Description

The contract owner has the authority to mint tokens. The owner may take advantage of it by calling the `mint` function. As a result the total supply will be highly inflated with new tokens.

```
function mint(address to, uint256 amount) public onlyOwner {  
    _mint(to, amount);  
}
```

Recommendation

The owner should carefully manage the credentials of the owner's account. We advised considering an extra-strong security mechanism that the actions may be quarantined by many users instead of one. The owner could also renounce the contract ownership for a period of time or pass the access to the zero address.

BC - Blacklisted Contracts

Criticality	medium
Location	contract.sol#L1

Description

The contract owner has the authority to stop contracts from transactions. The owner may take advantage of it by calling the `enableBlacklist` function.

```
require(!isBlacklisted(msg.sender), "Wenabisv2: sender en lista negra");  
require(!isBlacklisted(recipient), "Wenabisv2n: receptor en blacklisted");  
require(!isBlacklisted(tx.origin), "Wenabisv2: sender en blacklisted");
```

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.

Contract Diagnostics

● Critical ● Medium ● Minor

Severity	Code	Description
●	L01	Public Function could be Declared External
●	L02	State Variables could be Declared Constant
●	L05	Unused State Variable
●	L04	Conformance to Solidity Naming Conventions
●	L09	Dead Code Elimination

L01 - Public Function could be Declared External

Criticality

minor

Location

@openzeppelin/contracts/access/Ownable.sol#L331,L327,L128 and 15 more

Description

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

```
mint
unpause
disableBlacklist
...
```

Recommendation

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

L02 - State Variables could be Declared Constant

Criticality

minor

Location

contracts/Wenabisv2.sol#L26,L25,L30 and 1 more

Description

Constant state variables should be declared constant to save gas.

```
_SYMBOL  
_NAME  
_MAX  
...
```

Recommendation

Add the constant attribute to state variables that never change.

L05 - Unused State Variable

Criticality

minor

Location

contracts/Wenabisv2.sol#L26,L25

Description

There are segments that contains unused state variable.

```
_SYMBOL  
_NAME
```

Recommendation

Remove unused state variables.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contracts/Wenabisv2.sol#L48,L47,L46 and 10 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.

```
ORIG_CHARITY_FEE  
ORIG_BURN_FEE  
ORIG_TAX_FEE  
...
```

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

@openzeppelin/contracts/utils/Address.sol#L320,L35,L47 and 20 more

Description

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

```
_getTaxFee  
trySub  
tryMul  
...
```

Recommendation

Remove unused functions.

Contract Functions

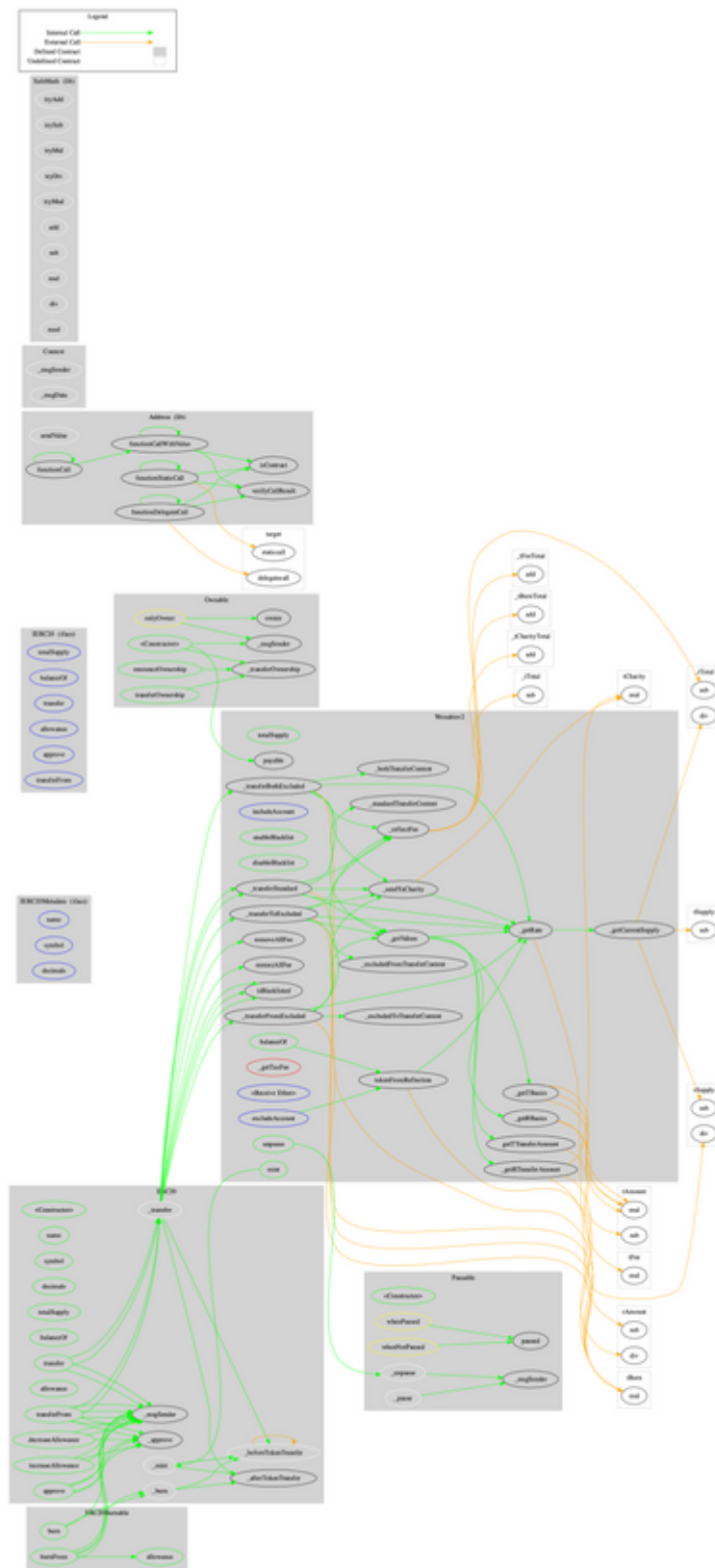
Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
Ownable	Implementation	Context		
	<Constructor>	Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
	_transferOwnership	Internal	✓	
Pausable	Implementation	Context		
	<Constructor>	Public	✓	-
	paused	Public		-
	_pause	Internal	✓	whenNotPaused
	_unpause	Internal	✓	whenPaused
ERC20	Implementation	Context, IERC20, IERC20Metadata		
	<Constructor>	Public	✓	-
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	increaseAllowance	Public	✓	-

	decreaseAllowance	Public	✓	-
	_transfer	Internal	✓	
	_mint	Internal	✓	
	_burn	Internal	✓	
	_approve	Internal	✓	
	_beforeTokenTransfer	Internal	✓	
	_afterTokenTransfer	Internal	✓	
ERC20Burnable	Implementation	Context, ERC20		
	burn	Public	✓	-
	burnFrom	Public	✓	-
IERC20Metadata	Interface	IERC20		
	name	External		-
	symbol	External		-
	decimals	External		-
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
Address	Library			
	isContract	Internal		
	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionStaticCall	Internal		
	functionStaticCall	Internal		

	functionDelegateCall	Internal	✓	
	functionDelegateCall	Internal	✓	
	verifyCallResult	Internal		
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
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		
Wenabisv2	Implementation	ERC20, Ownable, Pausable, ERC20Burn able		
	<Constructor>	Public	Payable	ERC20
	totalSupply	Public		-
	balanceOf	Public		-
	tokenFromReflection	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	excludeAccount	External	✓	onlyOwner

	includeAccount	External	✓	onlyOwner
	enableBlacklist	Public	✓	onlyOwner
	disableBlacklist	Public	✓	onlyOwner
	isBlacklisted	Public		-
	_transfer	Internal	✓	
	_transferStandard	Private	✓	
	_standardTransferContent	Private	✓	
	_transferToExcluded	Private	✓	
	_excludedFromTransferContent	Private	✓	
	_transferFromExcluded	Private	✓	
	_excludedToTransferContent	Private	✓	
	_transferBothExcluded	Private	✓	
	_bothTransferContent	Private	✓	
	_reflectFee	Private	✓	
	_getValues	Private		
	_getTBasics	Private		
	getTTransferAmount	Private		
	_getRBasics	Private		
	_getRTransferAmount	Private		
	_getRate	Private		
	_getCurrentSupply	Private		
	_sendToCharity	Private	✓	
	removeAllFee	Private	✓	
	restoreAllFee	Private	✓	
	_getTaxFee	Private		
	<Receive Ether>	External	Payable	-
	unpause	Public	✓	onlyOwner
	mint	Public	✓	onlyOwner
	_beforeTokenTransfer	Internal	✓	whenNotPaused

Contract Flow



Domain Info

Domain Name	wenabis.co
Registry Domain ID	D2BEC39862EE14CFDAAD19477EE0D6C32-NSR
Creation Date	2021-05-06T03:33:40Z
Updated Date	2021-07-17T01:32:25Z
Registry Expiry Date	2022-05-06T03:33:40Z
Registrar WHOIS Server	whois.godaddy.com
Registrar URL	whois.godaddy.com
Registrar	GoDaddy.com, LLC
Registrar IANA ID	146

The domain has been created 9 months before the creation of the audit. It will expire in 3 months.

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

Summary

Wenabis claims to be a utility token based on the research, exploration, and industrial production of the Cannabis and Cannabinoids for medicinal purposes and social impact.

The Project has a friendly and growing community. The Smart Contract analysis reported no compiler error, 1 critical and 1 medium threat issues. There are some functions that can be abused by the owner, like minting new tokens and blacklisting wallets from transactions. 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.

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

About Coinscope

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.

Coinscope is aiming to make crypto discoverable and efficient globally. It provides all the essential tools to assist users draw their own conclusions.



The Coinscope.co team

<https://www.coinscope.co>