Cyberscope

Audit Report

RAISER NETWORK

June 2022

Type	BEP20
Network	BSC
Address	0x9706d6eeb90ff975a10b6823d6aa5e3434e54a8e
Audited by	© cyberscope

Table of Contents

Contract Review

Contract Name	RaiserToken
Compiler Version	v0.8.2+commit.661d1103
Optimization	200 runs
Licence	None
Explorer	https://bscscan.com/token/0x9706d6EeB90ff975A10b6823d6aA5E3434E54a8E
Symbol	RAIZ
Decimals	18
Total Supply	100,000,000
Domain	raiser.network

Source Files

Filename	**SHA256**
contract.sol	b8a5e88b391a9ad358f8513e295e7815d1fbf122cd9bf8f63647200a85636d6d

Audit Updates

Initial Audit	1st June 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
	L04	Conformance to Solidity Naming Conventions
	L05	Unused State Variable
	L09	Dead Code Elimination
	L12	Using Variables before Declaration
	L14	Uninitialized Variables in Local Scope
	L15	Local Scope Variable Shadowing

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contract.sol#L764,1095,1099,1146,1156,1077,1078,1165,1187,1190,1205,1378,1382,1717,1756,1761,1799,1819,1749,1851,1854,2083

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.

```
__gap
__ERC20Votes_init_unchained
__ERC20Votes_init
_PERMIT_TYPEHASH
DOMAIN_SEPARATOR
__ERC20Permit_init_unchained
__ERC20Permit_init
__ERC20_init_unchained
__ERC20_init
...
```

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#L2083

Description

There are segments that contain unused state variables.

```
__gap
```

Recommendation

Remove unused state variables.

L09 - Dead Code Elimination

Criticality	minor
Location	contract.sol#L853,863,882,896,915,925,828,942,1187,1190,389,397,574,602,686,672,529,587,1095,1608,1851,1854,2070,2000,2074,350,324,331,2126,155,209,239,191,173,227,50,110,137,80,125,65,443,459,418

Description

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

```
toString
toHexString
toUint96
toUint8
toUint64
toUint256
toUint16
toUint128
toInt8
...
```

Recommendation

Remove unused functions.

L12 - Using Variables before Declaration

Criticality	minor
Location	contract.sol#L534,2043

Description

The contract is using a variable before the declaration. This is usually happening either if it has not been declared yet or the variable has been declared in a different scope.

```
oldWeight
newWeight
r
```

Recommendation

The variables should be declared before any usage of them.

L14 - Uninitialized Variables in Local Scope

Criticality	minor
Location	contract.sol#L2048

Description

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

```
newWeight_scope_1
oldWeight_scope_0
```

Recommendation

All the local scoped variables should be initialized.

L15 - Local Scope Variable Shadowing

Criticality	minor
Location	contract.sol#L1756

Description

The are variables that are defined in the local scope containing the same name from an upper scope.

```
name
```

Recommendation

The local variables should have different names from the upper scoped variables.

Contract Functions

Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
SafeCastUpgradeable	Library			
	toUint224	Internal		
	toUint128	Internal		
	toUint96	Internal		
	toUint64	Internal		
	toUint32	Internal		
	toUint16	Internal		
	toUint8	Internal		
	toUint256	Internal		
	toInt128	Internal		
	toInt64	Internal		
	toInt32	Internal		
	toInt16	Internal		
	toInt8	Internal		
	toInt256	Internal		
IVotesUpgradeable	Interface			
	getVotes	External		-
	getPastVotes	External		-
	getPastTotalSupply	External		-
	delegates	External		-
	delegate	External	✓	-
	delegateBySig	External	✓	-
MathUpgradeable	Library			
	max	Internal		
	min	Internal		

	average	Internal		
	ceilDiv	Internal		
CountersUpgradeable	Library			
	current	Internal		
	increment	Internal	✓	
	decrement	Internal	✓	
	reset	Internal	✓	
StringsUpgradeable	Library			
	toString	Internal		
	toHexString	Internal		
	toHexString	Internal		
ECDSAUpgradeable	Library			
	_throwError	Private		
	tryRecover	Internal		
	recover	Internal		
	tryRecover	Internal		
	recover	Internal		
	tryRecover	Internal		
	recover	Internal		
	toEthSignedMessageHash	Internal		
	toEthSignedMessageHash	Internal		
	toTypedDataHash	Internal		
IERC20PermitUpgradeable	Interface			
	permit	External	✓	-
	nonces	External		-
	DOMAIN_SEPARATOR	External		-
AddressUpgradeable	Library			

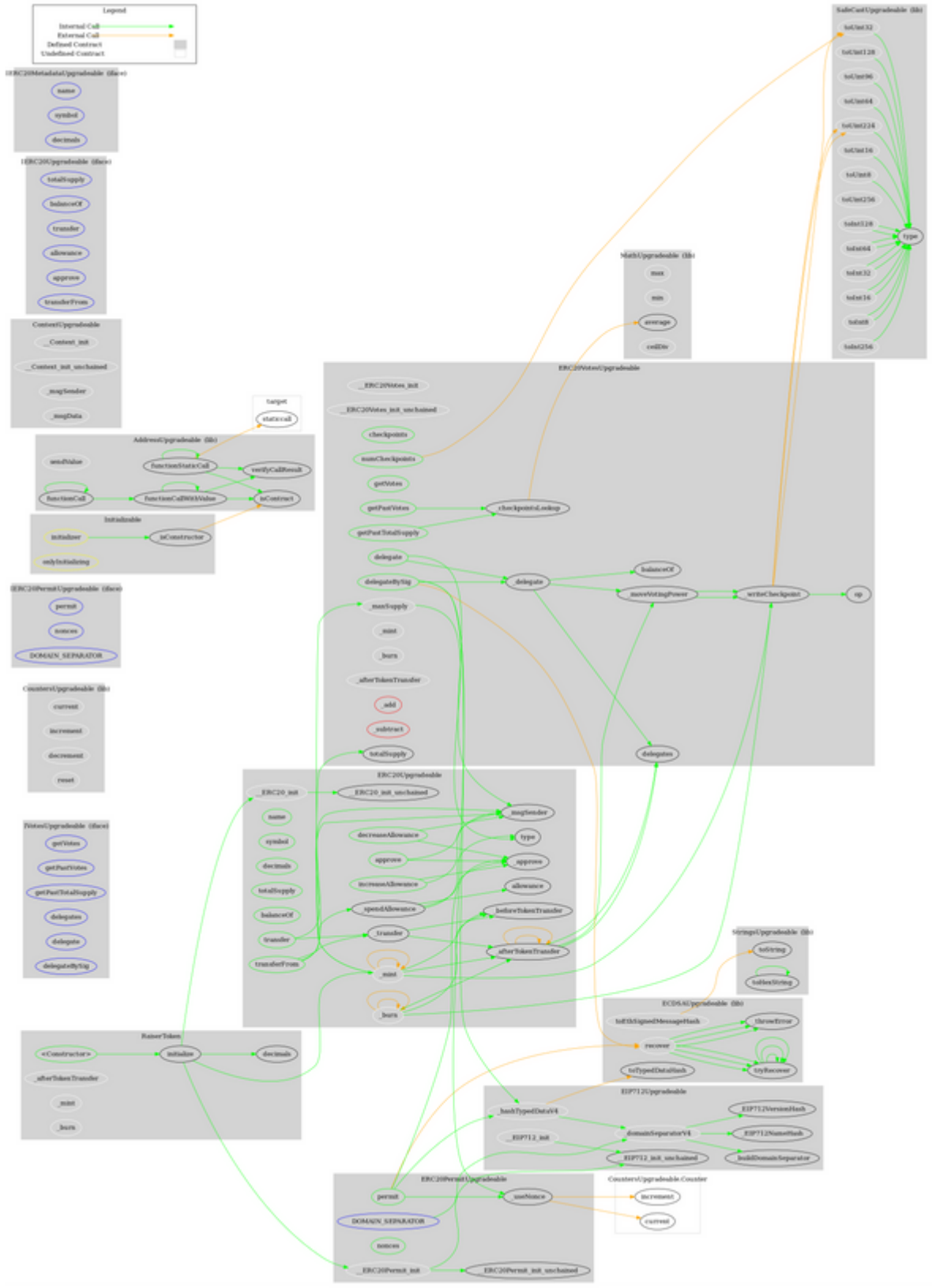
	isContract	Internal		
	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionStaticCall	Internal		
	functionStaticCall	Internal		
	verifyCallResult	Internal		
Initializable	Implementation			
	_isConstructor	Private		
EIP712Upgradeable	Implementation	Initializable		
	__EIP712_init	Internal	✓	onlyInitializing
	__EIP712_init_unchained	Internal	✓	onlyInitializing
	_domainSeparatorV4	Internal		
	_buildDomainSeparator	Private		
	_hashTypedDataV4	Internal		
	_EIP712NameHash	Internal		
	_EIP712VersionHash	Internal		
ContextUpgradeable	Implementation	Initializable		
	__Context_init	Internal	✓	onlyInitializing
	__Context_init_unchained	Internal	✓	onlyInitializing
	_msgSender	Internal		
	_msgData	Internal		
IERC20Upgradeable	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-

	approve	External	✓	-
	transferFrom	External	✓	-
IERC20MetadataUpgradeable	Interface	IERC20Upgradeable		
	name	External		-
	symbol	External		-
	decimals	External		-
ERC20Upgradeable	Implementation	Initializable, ContextUpgradeable, IERC20Upgradeable, IERC20MetadataUpgradeable		
	__ERC20_init	Internal	✓	onlyInitializing
	__ERC20_init_unchained	Internal	✓	onlyInitializing
	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	✓	
	_spendAllowance	Internal	✓	
	_beforeTokenTransfer	Internal	✓	
	_afterTokenTransfer	Internal	✓	

ERC20PermitUpgradeable	Implementation	Initializable, ERC20Upgradeable, IERC20PermitUpgradeable, EIP712Upgradeable		
	__ERC20Permit_init	Internal	✓	onlyInitializing
	__ERC20Permit_init_unchained	Internal	✓	onlyInitializing
	permit	Public	✓	-
	nonces	Public		-
	DOMAIN_SEPARATOR	External		-
	_useNonce	Internal	✓	
ERC20VotesUpgradeable	Implementation	Initializable, IVotesUpgradeable, ERC20PermitUpgradeable		
	__ERC20Votes_init	Internal	✓	onlyInitializing
	__ERC20Votes_init_unchained	Internal	✓	onlyInitializing
	checkpoints	Public		-
	numCheckpoints	Public		-
	delegates	Public		-
	getVotes	Public		-
	getPastVotes	Public		-
	getPastTotalSupply	Public		-
	_checkpointsLookup	Private		
	delegate	Public	✓	-
	delegateBySig	Public	✓	-
	_maxSupply	Internal		
	_mint	Internal	✓	
	_burn	Internal	✓	
	_afterTokenTransfer	Internal	✓	
	_delegate	Internal	✓	
	_moveVotingPower	Private	✓	

	_writeCheckpoint	Private	✓	
	_add	Private		
	_subtract	Private		
RaiserToken	Implementation	Initializable, ERC20Upgradeable, ERC20PermitUpgradeable, ERC20VotesUpgradeable		
	<Constructor>	Public	✓	initializer
	initialize	Public	✓	initializer
	_afterTokenTransfer	Internal	✓	
	_mint	Internal	✓	
	_burn	Internal	✓	

Contract Flow



Domain Info

Domain Name	raiser.network
Registry Domain ID	53fd5537b785497abd0de6e3cb220801-DONUTS
Creation Date	2022-01-20T23:16:22Z
Updated Date	2022-03-14T20:55:22Z
Registry Expiry Date	2023-01-20T23:16:22Z
Registrar WHOIS Server	whois.namesilo.com
Registrar URL	http://www.namesilo.com
Registrar	NameSilo, LLC
Registrar IANA ID	1479

The domain has been created 4 months before the creation of the audit. It will expire in 8 months.

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

Summary

RAISER NETWORK 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 access some admin functions that can not be used in a malicious way to disturb the users' transactions.

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