



Cyberscope

Audit Report

GUTS INU

April 2022

Type BEP20

Network BSC

Address 0x498b70f141763B1f06B33C50E9Ae5cD8f1058e31

Audited by © cyberscope

Table of Contents

Table of Contents	1
Contract Review	3
Audit Updates	3
Source Files	4
Contract Analysis	5
ELFM - Exceed Limit Fees Manipulation	6
Description	6
Recommendation	6
Contract Diagnostics	7
L01 - Public Function could be Declared External	8
Description	8
Recommendation	8
L02 - State Variables could be Declared Constant	9
Description	9
Recommendation	9
L04 - Conformance to Solidity Naming Conventions	10
Description	10
Recommendation	10
L07 - Missing Events Arithmetic	11
Description	11
Recommendation	11
L13 - Divide before Multiply Operation	12
Description	12
Recommendation	12
L15 - Local Scope Variable Shadowing	13
Description	13

Recommendation	13
Contract Functions	14
Contract Flow	18
Domain Info	19
Summary	20
Disclaimer	21
About Cyberscope	22

Contract Review

Contract Name	GUTSINUTK
Compiler Version	v0.8.7+commit.e28d00a7
Optimization	200 runs
Licence	MIT
Explorer	https://bscscan.com/token/0x498b70f141763B1f06B33C50E9Ae5cD8f1058e31
Symbol	GUTS
Decimals	18
Total Supply	1,000,000,000
Domain	gutsinu.net

Audit Updates

Initial Audit	18th April 2022
Corrected	

Source Files

Filename	SHA256
Address.sol	95ec5abe632a8bbde430a5cd751b7c862791898675e0d0e8862caf2084ecd1d9
Context.sol	89cf3ea0215029b05f335b893845bfcfe3d32860a3ade8808fdd4ba02114e81c
ERC20.sol	1e0d13cffbf30acc1e8b5bf29456cd950b268d7576d4e1cd653d01f94c07b3ee
GUTSINUTK.sol	d2920f664f12de89e2f9fab7448fdc1d808f239d06e3bf736b9913b8e77ede68
IERC20.sol	f4cd84fe07f21266ffaaf108bbaef56d6679af12ef13f53b5a12a42b4b1d1381
IUniswapV2Factory.sol	2481db85ce86dd6c0aa7e2d69030b1e6b18d376ebf0b7d0e9085af860fb7d40d
IUniswapV2Router01.sol	5563ddc2f75c2f6f31bc38b57794ceff4ea33de9f6fecb5d037f0691f16d53b2
IUniswapV2Router02.sol	3e72e65e02e79fbafe86b01c0c98fed4cfc65d67e30f7e0210eee1a0b485d3ff
Ownable.sol	613ab99c1d6f98fbdfda8e1c8038f395f25f8d5851e67593bd77cea71a35ba1f
SafeMath.sol	f4b9881ce06892493d5d14743b0a9f90c8f8fae489baee7d08b8b2a496976549

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

ELFM - Exceed Limit Fees Manipulation

Criticality	critical
Location	contract.sol#L50

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 `setTransactionFee` function with a high percentage value.

```
function setTransactionFee(uint256 fee) public onlyOwner {  
    _taxfee = fee;  
}
```

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
●	L01	Public Function could be Declared External
●	L02	State Variables could be Declared Constant
●	L04	Conformance to Solidity Naming Conventions
●	L07	Missing Events Arithmetic
●	L13	Divide before Multiply Operation
●	L15	Local Scope Variable Shadowing

L01 - Public Function could be Declared External

Criticality	minor
Location	GUTSINUTK.sol#L43,50

Description

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

```
setTransactionFee  
setFeeWallet
```

Recommendation

Use the external attribute for functions never called from the contract

L02 - State Variables could be Declared Constant

Criticality

minor

Location

GUTSINUTK.sol#L20

Description

Constant state variables should be declared constant to save gas.

```
TOTAL_SUPPLY
```

Recommendation

Add the constant attribute to state variables that never change.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	GUTSINUTK.sol#L26,27,28,29,15,16,17,20

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.

```
TOTAL_SUPPLY  
_feeWallet  
_taxfee  
_isInit  
_burn  
_team  
_marketingAddr  
_preSale
```

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

GUTSINUTK.sol#L50

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.

```
_taxfee = fee
```

Recommendation

Emit an event for critical parameter changes.

L13 - Divide before Multiply Operation

Criticality

minor

Location

GUTSINUTK.sol#L54

Description

Performing divisions before multiplications may cause lose of prediction.

```
_mfee = amount.div(100).mul(_taxfee)
```

Recommendation

The multiplications should be prior to the divisions.

L15 - Local Scope Variable Shadowing

Criticality

minor

Location

GUTSINUTK.sol#L21,29

Description

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

```
_burn  
_symbol  
_name
```

Recommendation

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

Contract Functions

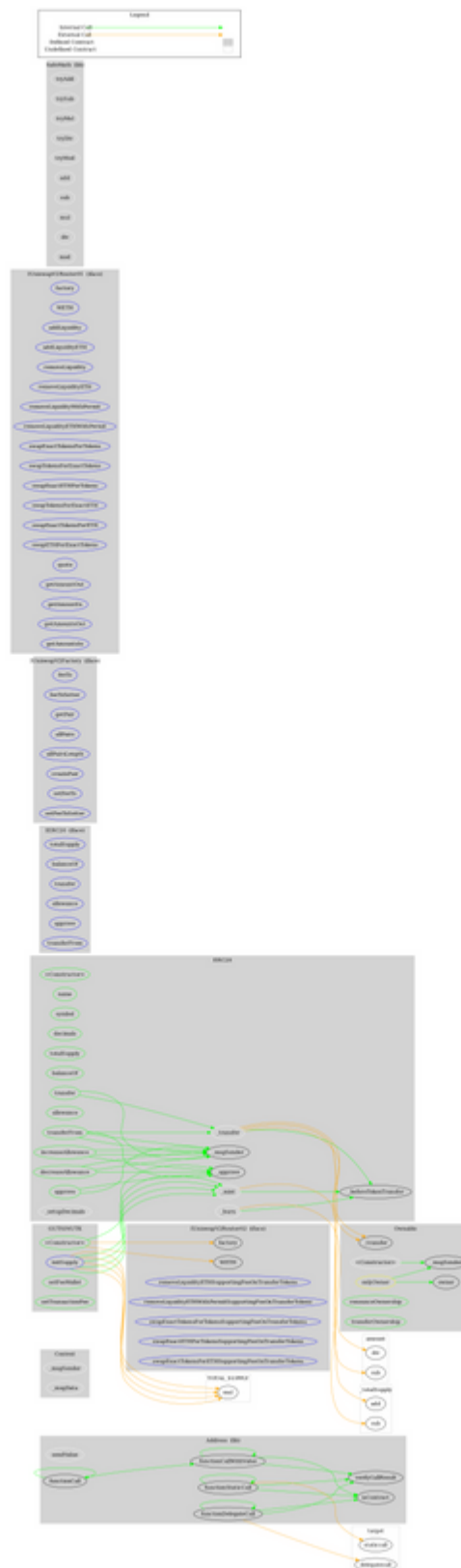
Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
Address	Library			
	isContract	Internal		
	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionStaticCall	Internal		
	functionStaticCall	Internal		
	functionDelegateCall	Internal	✓	
	functionDelegateCall	Internal	✓	
	verifyCallResult	Internal		
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
ERC20	Implementation	Context, IERC20		
	<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	✓	
	_setupDecimals	Internal	✓	
	_beforeTokenTransfer	Internal	✓	
GUTSINUTK	Implementation	ERC20, Ownable		
	<Constructor>	Public	✓	ERC20
	initSupply	External	✓	onlyOwner
	setFeeWallet	Public	✓	onlyOwner
	setTransactionFee	Public	✓	onlyOwner
	_transfer	Internal	✓	
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
IUniswapV2Factory	Interface			
	feeTo	External		-
	feeToSetter	External		-
	getPair	External		-
	allPairs	External		-
	allPairsLength	External		-
	createPair	External	✓	-
	setFeeTo	External	✓	-
	setFeeToSetter	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	✓	-
Ownable	Implementation	Context		
	<Constructor>	Internal	✓	
	owner	Public		-

	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
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		

Contract Flow



Domain Info

Domain Name	gutsinu.net
Registry Domain ID	2689835233_DOMAIN_NET-VRSN
Creation Date	2022-04-17T05:19:38Z
Updated Date	2022-04-17T05:19:39Z
Registry Expiry Date	2023-04-17T05:19:38Z
Registrar WHOIS Server	whois.hostinger.com
Registrar URL	http://www.hostinger.com
Registrar	Hostinger, UAB
Registrar IANA ID	1636

The domain has been created 1 day 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

GUTS INU is an interesting project that has a friendly and growing community. The Smart Contract analysis reported one critical issue. The contract owner can manipulate the fees. 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 provides all the essential tools to assist users draw their own conclusions.



The Cyberscope team

<https://www.cyberscope.io>