



Audit Report

Meme Machine

January 2022

Type	BEP20
Network	BSC
Address	0xe340aADC9DB4a73B94b13e299c26E6C8C89AcD35
Audited by	© coinscope

Table of Contents

Table of Contents	1
Contract Review	3
Audit Updates	3
Contract Analysis	4
ELFM - Exceed Limit Fees Manipulation	5
Description	5
Recommendation	5
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
L05 - Unused State Variable	10
Description	10
Recommendation	10
L04 - Conformance to Solidity Naming Conventions	11
Description	11
Recommendation	11
L09 - Dead Code Elimination	12
Description	12
Recommendation	12
L07 - Missing Events Arithmetic	13
Description	13
Recommendation	13

Contract Functions	14
Contract Flow	20
Domain Info	21
Summary	22
Disclaimer	23
About Coinscope	24

Contract Review

Contract Name	MemeMachine
Compiler Version	v0.8.9+commit.e5eed63a
Optimization	200 runs
Licence	MIT
Explorer	https://bscscan.com/token/0xe340aADC9DB4a73B94b13e299c26E6C8C89AcD35
Symbol	MeMa
Decimals	18
Total Supply	1,000,000,000,000,000
Source	contract.sol
Domain	mematoken.io

Audit Updates

Initial Audit	24th 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 (10%)
●	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	medium
Location	contract.sol#L1428,L1077

Description

The contract owner has the authority to increase over the allowed limit of 10%. The owner may take advantage of it by calling the `setSellFee` function with a high percentage value.

```
function setSellFee(uint256 sellTaxFee, uint256 sellLiquidityFee, uint256
sellMarketingFee, uint256 sellTeamFee)
    external
    onlyOwner
{
    _sellTaxFee = sellTaxFee;
    _sellLiquidityFee = sellLiquidityFee;
    _sellMarketingFee = sellMarketingFee;
    _sellTeamFee = sellTeamFee;
    require(_sellTaxFee + _sellLiquidityFee + _sellMarketingFee + _sellTeamFee
<= 30, "Must keep taxes below 30%");
    emit SetSellFee(sellMarketingFee, sellLiquidityFee, sellTaxFee,
sellTeamFee);
}
```

Users that bought at the same time that the trading opens, will pay three times more fees. This penalty will be active three days after the trade opening.

```
if(boughtEarly[from] && earlyBuyPenaltyEnd <= block.number){
    _taxFee = _taxFee * 3;
    _liquidityFee = _liquidityFee * 3;
}
```

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
●	L05	Unused State Variable
●	L04	Conformance to Solidity Naming Conventions
●	L09	Dead Code Elimination
●	L07	Missing Events Arithmetic

L01 - Public Function could be Declared External

Criticality	minor
Location	contract.sol#L1462,L257,L253

Description

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

```
setSwapAndLiquifyEnabled  
getTime  
getUnlockTime
```

Recommendation

Use the external attribute for functions never called from the contract

L02 - State Variables could be Declared Constant

Criticality

minor

Location

contract.sol#L216,L217,L684

Description

Constant state variables should be declared constant to save gas.

```
_previousOwner  
_lockTime  
minimumTokensBeforeSwap
```

Recommendation

Add the constant attribute to state variables that never change.

L05 - Unused State Variable

Criticality	minor
Location	contract.sol#L216

Description

There are segments that contains unused state variable.

```
_previousOwner
```

Recommendation

Remove unused state variables.

L04 - Conformance to Solidity Naming Conventions

Criticality

minor

Location

contract.sol#L678,L663,L662 and 25 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.

```
_isExcludedMaxTransactionAmount  
_sellTeamFee  
_sellMarketingFee  
...
```

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

contract.sol#L108,L104,L18 and 7 more

Description

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

```
mod
_msgData
sendValue
...
```

Recommendation

Remove unused functions.

L07 - Missing Events Arithmetic

Criticality

minor

Location

contract.sol#L916

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.

```
gasPriceLimit = gas * 1000000000
```

Recommendation

Emit an event for critical parameter changes.

Contract Functions

Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
SafeMath	Library			
	add	Internal		
	sub	Internal		
	sub	Internal		
	mul	Internal		
	div	Internal		
	div	Internal		
	mod	Internal		
	mod	Internal		
Address	Library			
	isContract	Internal		
	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	Internal	✓	

	_functionCallWithValue	Private	✓	
Ownable	Implementation	Context		
	<Constructor>	Public	✓	-
	owner	Public		-
	renounceOwnership	External	✓	onlyOwner
	transferOwnership	External	✓	onlyOwner
	getUnlockTime	Public		-
	getTime	Public		-
IUniswapV2Factory	Interface			
	feeTo	External		-
	feeToSetter	External		-
	getPair	External		-
	allPairs	External		-
	allPairsLength	External		-
	createPair	External	✓	-
	setFeeTo	External	✓	-
	setFeeToSetter	External	✓	-
IUniswapV2Pair	Interface			
	name	External		-
	symbol	External		-
	decimals	External		-
	totalSupply	External		-
	balanceOf	External		-
	allowance	External		-
	approve	External	✓	-
	transfer	External	✓	-
	transferFrom	External	✓	-
	DOMAIN_SEPARATOR	External		-
	PERMIT_TYPEHASH	External		-
	nonces	External		-
	permit	External	✓	-

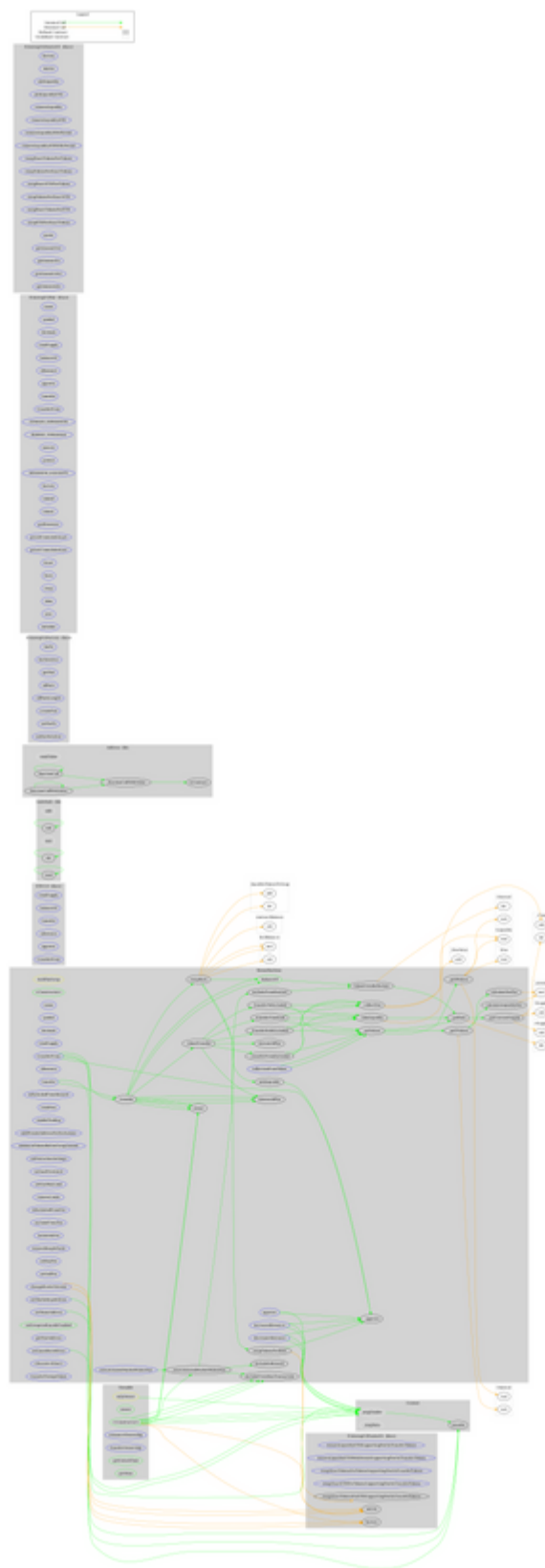
	MINIMUM_LIQUIDITY	External		-
	factory	External		-
	token0	External		-
	token1	External		-
	getReserves	External		-
	price0CumulativeLast	External		-
	price1CumulativeLast	External		-
	kLast	External		-
	burn	External	✓	-
	swap	External	✓	-
	skim	External	✓	-
	sync	External	✓	-
	initialize	External	✓	-
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	✓	-
MemeMachine	Implementation	Context, IERC20, Ownable		
	<Constructor>	Public	✓	-
	name	External		-
	symbol	External		-
	decimals	External		-
	totalSupply	External		-
	balanceOf	Public		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
	increaseAllowance	External	✓	-
	decreaseAllowance	External	✓	-
	isExcludedFromReward	External		-
	totalFees	External		-
	enableTrading	External	✓	onlyOwner
	addPresaleAddressForExclusions	External	✓	onlyOwner
	minimumTokensBeforeSwapAmount	External		-
	setAutomatedMarketMakerPair	External	✓	onlyOwner
	_setAutomatedMarketMakerPair	Private	✓	
	setProtectionSettings	External	✓	onlyOwner
	setGasPriceLimit	External	✓	onlyOwner
	setGasMaxLimit	External	✓	onlyOwner

	removeLimits	External	✓	onlyOwner
	reflectionFromToken	External		-
	tokenFromReflection	Public		-
	excludeFromReward	Public	✓	onlyOwner
	includeInReward	Public	✓	onlyOwner
	excludeFromMaxTransaction	Public	✓	onlyOwner
	_approve	Private	✓	
	_transfer	Private	✓	
	swapBack	Private	✓	lockTheSwap
	swapTokensForBNB	Private	✓	
	addLiquidity	Private	✓	
	_tokenTransfer	Private	✓	
	_transferStandard	Private	✓	
	_transferToExcluded	Private	✓	
	_transferFromExcluded	Private	✓	
	_transferBothExcluded	Private	✓	
	_reflectFee	Private	✓	
	_getValues	Private		
	_getTValues	Private		
	_getRValues	Private		
	_getRate	Private		
	_getCurrentSupply	Private		
	_takeLiquidity	Private	✓	
	calculateTaxFee	Private		
	calculateLiquidityFee	Private		
	removeAllFee	Private	✓	
	restoreAllFee	Private	✓	
	isExcludedFromFee	External		-
	excludeFromFee	External	✓	onlyOwner
	includeInFee	External	✓	onlyOwner
	removeBoughtEarly	External	✓	onlyOwner
	setBuyFee	External	✓	onlyOwner
	setSellFee	External	✓	onlyOwner
	setMarketingAddress	External	✓	onlyOwner
	setTeamAddress	External	✓	onlyOwner

	setLiquidityAddress	External	✓	onlyOwner
	setSwapAndLiquifyEnabled	Public	✓	onlyOwner
	getPairAddress	External		onlyOwner
	changeRouterVersion	External	✓	onlyOwner
	<Receive Ether>	External	Payable	-
	transferForeignToken	External	✓	onlyOwner

Contract Flow



Domain Info

Domain Name	mematoken.io
Registry Domain ID	9b7f6b1a430143f3a7cef1415c48753d-DONUTS
Creation Date	2021-12-12T22:12:11Z
Updated Date	2021-12-17T22:12:54Z
Registry Expiry Date	2022-12-12T22:12:11Z
Registrar WHOIS Server	whois.domrobot.com
Registrar URL	https://www.inwx.com
Registrar	INWX GmbH & Co. KG
Registrar IANA ID	1420

The domain has been created about 1 month before the creation of the audit. It will expire in 11 months.

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

Summary

Meme Machine is a meme token. It has a friendly and growing community. The contract analysis reported one medium threat issue. The contract owner can potentially set the total fees to 30%. It could have a check for a more reasonable maximum value like 10%. 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>