



# Audit Report

## **Honshu Inu**

January 2022

Type           BEP20

Network       BSC

Address       0xc5e12d5BF85C5B2Cf968540cE57BD62658b37D0C

Audited by   © coinscope

# Table of Contents

<b>Table of Contents</b>	<b>1</b>
<b>Contract Review</b>	<b>3</b>
<b>Audit Updates</b>	<b>3</b>
<b>Contract Analysis</b>	<b>4</b>
<b>ST - Stop Transactions</b>	<b>5</b>
Description	5
Recommendation	5
<b>ULTW - Unlimited Liquidity to Team Wallet</b>	<b>6</b>
Description	6
Recommendation	6
<b>Contract Diagnostics</b>	<b>7</b>
<b>L01 - Public Function could be Declared External</b>	<b>8</b>
Description	8
Recommendation	8
<b>L02 - State Variables could be Declared Constant</b>	<b>9</b>
Description	9
Recommendation	9
<b>L04 - Conformance to Solidity Naming Conventions</b>	<b>10</b>
Description	10
Recommendation	10
<b>L09 - Dead Code Elimination</b>	<b>11</b>
Description	11
Recommendation	11
<b>L07 - Missing Events Arithmetic</b>	<b>12</b>
Description	12
Recommendation	12

<b>Contract Functions</b>	<b>13</b>
<b>Contract Flow</b>	<b>19</b>
<b>Domain Info</b>	<b>20</b>
<b>Summary</b>	<b>21</b>
<b>Disclaimer</b>	<b>22</b>
<b>About Coinscope</b>	<b>23</b>

## Contract Review

<b>Contract Name</b>	HonshuInu
<b>Compiler Version</b>	v0.6.12+commit.27d51765
<b>Optimization</b>	200 runs
<b>Licence</b>	None
<b>Explorer</b>	<a href="https://bscscan.com/token/0xc5e12d5BF85C5B2Cf968540cE57BD62658b37D0C">https://bscscan.com/token/0xc5e12d5BF85C5B2Cf968540cE57BD62658b37D0C</a>
<b>Symbol</b>	HONSHU
<b>Decimals</b>	9
<b>Total Supply</b>	100,000,000,000,000,000
<b>Source</b>	contract.sol
<b>Domain</b>	honshuinu.io

## Audit Updates

<b>Initial Audit</b>	14th January 2022
<b>Corrected</b>	

# Contract Analysis

● Critical
 ● Medium
 ● Minor
 ● Pass

Severity	Code	Description
<span style="color: orange;">●</span>	ST	Contract Owner is not able to stop or pause transactions
<span style="color: blue;">●</span>	OCTD	Contract Owner is not able to transfer tokens from specific address
<span style="color: blue;">●</span>	OTUT	Owner Transfer User's Tokens
<span style="color: blue;">●</span>	ELFM	Contract Owner is not able to increase fees more than a reasonable percent (25%)
<span style="color: grey;">●</span>	ULTW	Contract Owner is not able to increase the amount of liquidity taken by dev wallet more than a reasonable percent
<span style="color: blue;">●</span>	MT	Contract Owner is not able to mint new tokens
<span style="color: blue;">●</span>	BT	Contract Owner is not able to burn tokens from specific wallet
<span style="color: blue;">●</span>	BC	Contract Owner is not able to blacklist wallets from selling

## ST - Stop Transactions

Criticality	medium
Location	contract.sol#L1231

### Description

The contract owner has the authority to stop transactions for all users excluding the owner. The owner may take advantage of it by setting the `_maxTxAmount` to zero.

```
if(from != owner() && to != owner())  
    require(amount <= _maxTxAmount, "Transfer amount exceeds the maxTxAmount.");
```

### Recommendation

The contract could embody a check for not allowing setting the `_maxTxAmount` less than a reasonable amount. A suggested implementation could check that the maximum amount should be more than a fixed percentage of the total supply.

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.

## ULTW - Unlimited Liquidity to Team Wallet

Criticality	minor
Location	contract.sol#L1354

### Description

The contract owner has the authority to transfer funds without limit to the team wallet. These funds have been swapped from the swap & liquify feature. The owner may take advantage of it by following the next steps:

1. Set the `_marketingFee` to 10 and all the other fees to 0
2. Trigger the `triggerSwapAndLiquify` setting `percent_Of_Tokens` to 100

```
function triggerSwapAndLiquify (uint256 percent_Of_Tokens) public onlyOwner {  
    // Do not trigger if already in swap  
    require(!inSwapAndLiquify, "Currently processing liquidity, try later.");  
    if (percent_Of_Tokens > 100){percent_Of_Tokens == 100;}  
    uint256 tokensOnContract = balanceOf(address(this));  
    uint256 sendTokens = tokensOnContract.mul(percent_Of_Tokens).div(100);  
    swapAndLiquify(sendTokens);  
}
```

### 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
●	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#L1354,L1211,L991 and 26 more

### Description

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

```
triggerSwapAndLiquify  
isExcludedFromFee  
includeInFee  
...
```

### Recommendation

Use the external attribute for functions never called from the contract

## L02 - State Variables could be Declared Constant

**Criticality**

minor

**Location**

contract.sol#L713,L712,L747 and 2 more

### Description

Constant state variables should be declared constant to save gas.

```
_symbol  
_name  
_dogeAddress  
...
```

### Recommendation

Add the constant attribute to state variables that never change.

## L04 - Conformance to Solidity Naming Conventions

**Criticality**

minor

**Location**

contract.sol#L760,L745,L744 and 32 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.

```
_maxTxAmount  
_swapForwardAddress  
_autoSellAddie  
...
```

### 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#L238,L222,L249 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#L1068,L1057,L1030 and 5 more

### 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.

```
numTokensSellToAddToLiquidity = _numTokensSellToAddToLiquidity
_maxTxAmount = _tTotal.mul(maxTxPercent).div(10 ** 2)
_burnFee = burnFee
...
```

### Recommendation

Emit an event for critical parameter changes.

# Contract Functions

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

	_functionCallWithValue	Private	✓	
<b>IUniswapV2Factory</b>	Interface			
	feeTo	External		-
	feeToSetter	External		-
	getPair	External		-
	allPairs	External		-
	allPairsLength	External		-
	createPair	External	✓	-
	setFeeTo	External	✓	-
	setFeeToSetter	External	✓	-
<b>IUniswapV2Pair</b>	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		-

	mint	External	✓	-
	burn	External	✓	-
	swap	External	✓	-
	skim	External	✓	-
	sync	External	✓	-
	initialize	External	✓	-
<b>IUniswapV2Router01</b>	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		-
<b>IUniswapV2Router02</b>	Interface	IUniswapV2 Router01		
	removeLiquidityETHSupportingFeeOnTransferTokens	External	✓	-
	removeLiquidityETHWithPermitSupportingFeeOnTransferTokens	External	✓	-
	swapExactTokensForTokensSupportingFeeOnTransferTokens	External	✓	-

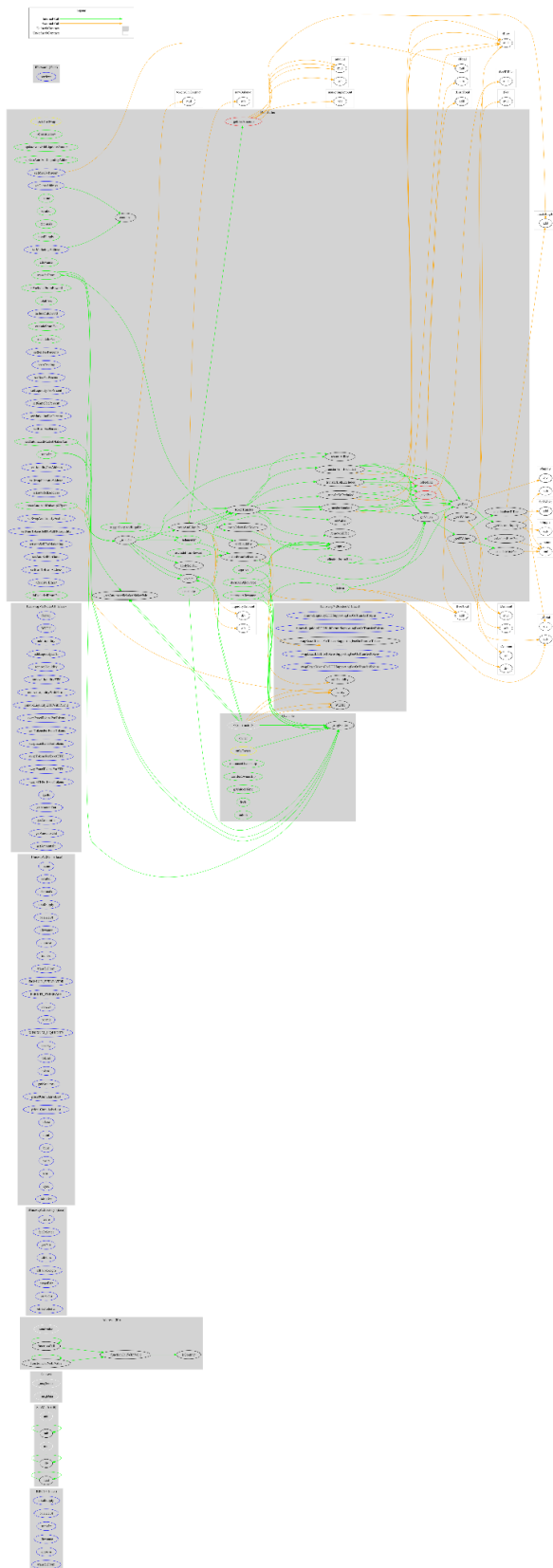


	swapExactETHForTokensSupporting FeeOnTransferTokens	External	Payable	-
	swapExactTokensForETHSupporting FeeOnTransferTokens	External	✓	-
<b>Ownable</b>	Implementation	Context		
	<Constructor>	Internal	✓	
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
	geUnlockTime	Public		-
	lock	Public	✓	onlyOwner
	unlock	Public	✓	-
<b>IForward</b>	Interface			
	retrieve	External	✓	-
<b>HonshuInu</b>	Implementation	Context, IERC20, Ownable		
	<Constructor>	Public	✓	-
	updateAutoAddLiquidityRouter	Public	✓	onlyOwner
	updateAutoAddLiquidityAddie	Public	✓	onlyOwner
	createAutoAddUniswapV2pair	Public	✓	onlyOwner
	setAutomatedMarketMakerPair	Public	✓	onlyOwner
	_setAutomatedMarketMakerPair	Private	✓	
	distribute	Public	✓	onlyOwner
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	setWallet	Public	✓	-
	contains	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-

	transferFrom	Public	✓	-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	isExcludedFromReward	Public		-
	totalFees	Public		-
	deliver	Public	✓	-
	reflectionFromToken	Public		-
	tokenFromReflection	Public		-
	excludeFromReward	Public	✓	onlyOwner
	includeInReward	External	✓	onlyOwner
	excludeFromFee	Public	✓	onlyOwner
	includeInFee	Public	✓	onlyOwner
	setSellFeePercents	External	✓	onlyOwner
	openTrading	External	✓	onlyOwner
	setTaxFeePercent	External	✓	onlyOwner
	setLiquidityFeePercent	External	✓	onlyOwner
	setGameFeePercent	External	✓	onlyOwner
	setMarketingFeePercent	External	✓	onlyOwner
	setBurnFeePercent	External	✓	onlyOwner
	setMarketingAddress	External	✓	onlyOwner
	setGameAddress	External	✓	onlyOwner
	setLiquidityTaxAddress	External	✓	onlyOwner
	setSwapForwardAddress	External	✓	onlyOwner
	setAutoSellAddress	External	✓	onlyOwner
	setMaxTxPercent	External	✓	onlyOwner
	setSwapAndLiquifyEnabled	External	✓	onlyOwner
	setNumTokensSellToAddToLiquidity	External	✓	onlyOwner
	setAutoSellForMarketing	External	✓	onlyOwner
	setAutoSellForGame	External	✓	onlyOwner
	setBurnToBurnAddress	External	✓	onlyOwner
	<Receive Ether>	External	Payable	-
	_reflectFee	Private	✓	
	_getValues	Private		
	_getTValues	Private		
	_getRValues	Private		

	_getRate	Private		
	_getCurrentSupply	Private		
	_takeFee	Private	✓	
	calculateTaxFee	Private		
	calculateBurnFee	Private		
	calculateFeeToTake	Private		
	removeAllFee	Private	✓	
	applySellFee	Private	✓	
	restoreAllFee	Private	✓	
	isExcludedFromFee	Public		-
	_approve	Private	✓	
	_transfer	Private	✓	
	_getFeeAmounts	Private		
	swapAndLiquify	Private	✓	lockTheSwap
	triggerSwapAndLiquify	Public	✓	onlyOwner
	swapTokensForTokens	Private	✓	
	addLiquidity	Private	✓	
	_tokenTransfer	Private	✓	
	_transferStandard	Private	✓	
	_transferToExcluded	Private	✓	
	_transferFromExcluded	Private	✓	
	_transferBothExcluded	Private	✓	

# Contract Flow



## Domain Info

<b>Domain Name</b>	honshuinu.io
<b>Registry Domain ID</b>	aa17a01965424a71af3e4ba5f2bd7b11-DONUTS
<b>Creation Date</b>	2022-01-04T01:01:08Z
<b>Updated Date</b>	2022-01-09T01:02:00Z
<b>Registry Expiry Date</b>	2023-01-04T01:01:08Z
<b>Registrar WHOIS Server</b>	whois.namecheap.com
<b>Registrar URL</b>	<a href="https://www.namecheap.com/">https://www.namecheap.com/</a>
<b>Registrar</b>	NameCheap, Inc.
<b>Registrar IANA ID</b>	1068

The domain has been created 10 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

Honshu Inu is a next generation decentralised metaverse meme game token built on Binance Smartchain. The token has a friendly and growing community. There are some functions that can be abused by the owner, like indirectly stopping the transactions and transferring funds to the team's wallet. 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>