Audit Report

ORDOCHAIN

February 2022

Type	BEP20
Network	BSC
Address	0x0106591B1372BbCcB9D2a5A1fe7f421c2D149C07
Audited by	© coinscope

Table of Contents

Contract Review

Contract Name	Ordo_Chain
Compiler Version	v0.8.11+commit.d7f03943
Optimization	200 runs
Licence	MIT
Explorer	https://bscscan.com/token/0x0106591B1372BbCcB9D2a5A1fe7f421c2D149C07
Symbol	ORDO
Decimals	9
Total Supply	1,000,000,000
Source	contract.sol
Domain	ordochain.com

Audit Updates

Initial Audit	10th February 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

ELFM - Exceed Limit Fees Manipulation

Criticality	critical
Location	contract.sol#L309

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 `setMarketingTax` function with a high percentage value.

```
  function setMarketingTax(uint256 percent_, address wallet_) public onlyOwner
{ iTaxPool.setMarketing(percent_, wallet_); }
```

```
 function setMarketing (uint256 percent_, address wallet_) public onlyOwner {
marketingPercentage = percent_; marketingWallet = wallet_; }
```

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.

ULTW - Unlimited Liquidity to Team Wallet

Criticality	medium
Location	contract.sol#L307

Description

The contract owner has the authority to transfer funds without limit to the team wallet. These funds have been accumulated from fees into the contract balance. The owner may take advantage of it by setting a high fee to the `marketingWallet` variable and then calling `distribute`.

```
  function distribute() public onlyOwner swapping {
    require(stakingrewardPercentage + marketingPercentage +
buybackandburnPercentage == maxPercent, "The sum of percentage isn't 100.");
    require(
      stakingrewardWallet != address(0)
      && marketingWallet != address(0)
      && buybackandburnWallet != address(0)
      ,
      "Cannot send to zero wallet."
    );
    uint256 amount = getBalance();
    (bool sent_1, ) = payable(stakingrewardWallet).call{value: (amount *
stakingrewardPercentage / maxPercent), gas: 30000}(""); require(sent_1,
"Transfer wallet_1 error."); balance = address(this).balance;
    (bool sent_2, ) = payable(marketingWallet).call{value: (amount *
marketingPercentage / maxPercent), gas: 30000}(""); require(sent_2, "Transfer
wallet_2 error."); balance = address(this).balance;
    (bool sent_3, ) = payable(buybackandburnWallet).call{value: (amount *
buybackandburnPercentage / maxPercent), gas: 30000}(""); require(sent_3,
"Transfer wallet_3 error."); balance = address(this).balance;
  }
```

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#L337,L333,L320 and 27 more

Description

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

```
decreaseAllowance
increaseAllowance
transferFrom
...
```

Recommendation

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

L02 - State Variables could be Declared Constant

Criticality	minor
Location	contract.sol#L200,L250,L249 and 6 more

Description

Constant state variables should be declared constant to save gas.

```
maxPercent
_symbol
_name
...
```

Recommendation

Add the constant attribute to state variables that never change.

L05 - Unused State Variable

Criticality	minor
Location	contract.sol#L267,L265

Description

There are segments that contains unused state variable.

```
FACTORY
ZERO
```

Recommendation

Remove unused state variables.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contract.sol#L276,L275,L270 and 13 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.

```
_excludedSellFee
_excludedBuyFee
_router
...
```

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#L431,L27

Description

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

```
_burn
_msgData
```

Recommendation

Remove unused functions.

L07 - Missing Events Arithmetic

Criticality	minor
Location	contract.sol#L257,L256

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.

```
_taxDivider = input_
_tax = input_
```

Recommendation

Emit an event for critical parameter changes.

Contract Functions

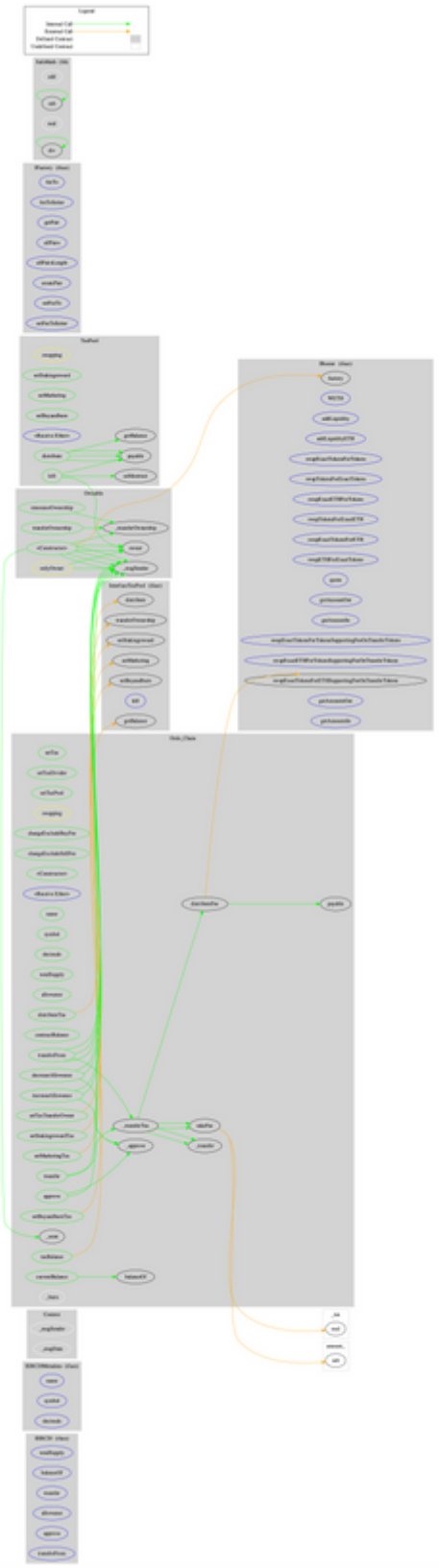
Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
IERC20Metadata	Interface	IERC20		
	name	External		-
	symbol	External		-
	decimals	External		-
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
Ownable	Implementation	Context		
	<Constructor>	Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
	_transferOwnership	Internal	✓	
IFactory	Interface			
	feeTo	External		-

	feeToSetter	External		-
	getPair	External		-
	allPairs	External		-
	allPairsLength	External		-
	createPair	External	✓	-
	setFeeTo	External	✓	-
	setFeeToSetter	External	✓	-
IRouter	Interface			
	factory	External		-
	WETH	External		-
	addLiquidity	External	✓	-
	addLiquidityETH	External	Payable	-
	swapExactTokensForTokens	External	✓	-
	swapTokensForExactTokens	External	✓	-
	swapExactETHForTokens	External	Payable	-
	swapTokensForExactETH	External	✓	-
	swapExactTokensForETH	External	✓	-
	swapETHForExactTokens	External	Payable	-
	quote	External		-
	getAmountOut	External		-
	getAmountIn	External		-
	swapExactTokensForTokensSupportingFeeOnTransferTokens	External	✓	-
	swapExactETHForTokensSupportingFeeOnTransferTokens	External	Payable	-
	swapExactTokensForETHSupportingFeeOnTransferTokens	External	✓	-
	getAmountsOut	External		-
	getAmountsIn	External		-
SafeMath	Library			
	add	Internal		
	sub	Internal		
	sub	Internal		
	mul	Internal		

	div	Internal		
	div	Internal		
TaxPool	Implementation	Ownable		
	setStakingreward	Public	✓	onlyOwner
	setMarketing	Public	✓	onlyOwner
	setBuyandburn	Public	✓	onlyOwner
	<Receive Ether>	External	Payable	-
	getBalance	Public		-
	distribute	Public	✓	onlyOwner swapping
	kill	Public	✓	onlyOwner
InterfaceTaxPool	Interface			
	setStakingreward	External	✓	-
	setMarketing	External	✓	-
	setBuyandburn	External	✓	-
	getBalance	External		-
	distribute	External	✓	-
	kill	External	✓	-
	transferOwnership	External	✓	-
Ordo_Chain	Implementation	Context, Ownable, IERC20, IERC20Metadata		
	setTax	Public	✓	onlyOwner
	setTaxDivider	Public	✓	onlyOwner
	setTaxPool	Public	✓	onlyOwner
	changeExcludeBuyFee	Public	✓	onlyOwner
	changeExcludeSellFee	Public	✓	onlyOwner
	<Constructor>	Public	✓	-
	<Receive Ether>	External	Payable	-
	name	Public		-
	symbol	Public		-

	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	allowance	Public		-
	currentBalance	Public		-
	contractBalance	Public		-
	taxBalance	Public		onlyOwner
	distributeTax	Public	✓	onlyOwner
	setStakingrewardTax	Public	✓	onlyOwner
	setMarketingTax	Public	✓	onlyOwner
	setBuyandburnTax	Public	✓	onlyOwner
	setTaxTransferOwner	Public	✓	onlyOwner
	transfer	Public	✓	-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	_transfer	Internal	✓	
	_transferTax	Internal	✓	
	takeFee	Private	✓	
	distributeFee	Private	✓	swapping
	_mint	Internal	✓	
	_burn	Internal	✓	
	_approve	Internal	✓	

Contract Flow



Domain Info

Domain Name	ordochain.com
Registry Domain ID	2673413545_DOMAIN_COM-VRSN
Creation Date	2022-02-06T21:30:58Z
Updated Date	2022-02-06T21:30:59Z
Registry Expiry Date	2023-02-06T21:30:58Z
Registrar WHOIS Server	whois.godaddy.com
Registrar URL	http://www.godaddy.com
Registrar	GoDaddy.com, LLC
Registrar IANA ID	146

The domain has been created 4 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

The Ordo aims to provide DeFi tools for token holders & businesses to buy, trade, create, and secure crypto assets with confidence. The Project has a friendly and growing community. There are some functions that can be abused by the owner, like manipulating fees 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