



# Audit Report

## **InuXV2**

February 2022

Type	BEP20
Network	BSC
Address	0xf9d6789b344645EcbFe833882f7C232a386bDFE2
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>ELFM - Exceed Limit Fees Manipulation</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>18</b>
<b>Domain Info</b>	<b>19</b>
<b>Summary</b>	<b>20</b>
<b>Disclaimer</b>	<b>21</b>
<b>About Coinscope</b>	<b>22</b>

## Contract Review

<b>Contract Name</b>	InuXV2
<b>Compiler Version</b>	v0.8.11+commit.d7f03943
<b>Optimization</b>	200 runs
<b>Licence</b>	None
<b>Explorer</b>	<a href="https://bscscan.com/token/0xf9d6789b344645EcbFe833882f7C232a386bDFE2">https://bscscan.com/token/0xf9d6789b344645EcbFe833882f7C232a386bDFE2</a>
<b>Symbol</b>	IXV2
<b>Decimals</b>	18
<b>Total Supply</b>	1,000,000,000
<b>Source</b>	contract.sol
<b>Domain</b>	inuxv2.com

## Audit Updates

<b>Initial Audit</b>	3rd February 2022
<b>Corrected</b>	

# Contract Analysis

● Critical
 ● Medium
 ● Minor
 ● Pass

Severity	Code	Description
<span style="color: red;">●</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: red;">●</span>	ELFM	Contract Owner is not able to increase fees more than a reasonable percent (25%)
<span style="color: blue;">●</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

<b>Criticality</b>	critical
<b>Location</b>	contract.sol#L448

### Description

The contract owner has the authority to stop the sales for all users excluding the owner. The owner may take advantage of it by setting a high value to `sellFeeMultiplier`. This action can happen only after the 57600th block number since the token's first transaction. Read more on the [fees manipulation finding](#).

```
} else if(isSell) {  
    applySellFeeMultiplier();  
}
```

### Recommendation

The contract could embody a check for not allowing setting the `sellFeeMultiplier` more 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.

## ELFM - Exceed Limit Fees Manipulation

<b>Criticality</b>	critical
<b>Location</b>	contract.sol#L407

### Description

The contract owner has the authority to increase the sale fees over the allowed limit of 25%. This can only happen after the 57600th block since the first transaction. The owner may take advantage of it by following the steps:

- Call `setBuyFees(0, 0, 0, 0, 5)`
- Call `setSellFeeMultiplier(4, 1)`
- Call `setBuyFees()` with the original fees

calling the `setTaxFeePercent` function with a high percentage value.

```
_taxFee *= sellFeeMultiplier / sellFeeDivisor;  
_liquidityFee *= sellFeeMultiplier / sellFeeDivisor;
```

### 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
●	L09	Dead Code Elimination
●	L07	Missing Events Arithmetic



## L01 - Public Function could be Declared External

<b>Criticality</b>	minor
<b>Location</b>	contract.sol#L351,L336,L331 and 20 more

### Description

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

```
setSwapSettings  
changeTeamWallet  
changeMarketingWallet  
...
```

### Recommendation

Use the external attribute for functions never called from the contract

## L02 - State Variables could be Declared Constant

<b>Criticality</b>	minor
<b>Location</b>	contract.sol#L130,L133,L132 and 1 more

### Description

Constant state variables should be declared constant to save gas.

```
_tTotal  
_symbol  
_name  
...
```

### Recommendation

Add the constant attribute to state variables that never change.

## L04 - Conformance to Solidity Naming Conventions

**Criticality**

minor

**Location**

contract.sol#L125,L365,L364 and 7 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.

```
_isExcludedFromTxLimits  
_amount  
_enabled  
...
```

### Recommendation

Follow the Solidity naming convention.

<https://docs.soliditylang.org/en/v0.4.25/style-guide.html#naming-conventions>

## L09 - Dead Code Elimination

<b>Criticality</b>	minor
<b>Location</b>	contract.sol#L26

### Description

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

```
_msgData
```

### Recommendation

Remove unused functions.

## L07 - Missing Events Arithmetic

**Criticality**

minor

**Location**

contract.sol#L351,L321,L307

### 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.

```
maxSwapAmount = newMaxSwapAmount * 10 ** 18  
sellFeeMultiplier = multiplier  
feeForLiquidity = liquidityFee
```

### 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>Context</b>	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
<b>Ownable</b>	Implementation	Context		
	<Constructor>	Public	✓	-
	owner	Public		-
	transferOwnership	Public	✓	onlyOwner
	_setOwner	Private	✓	
<b>IDexFactory</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>IDexRouter</b>	Interface			
	factory	External		-
	WETH	External		-
	addLiquidity	External	✓	-
	addLiquidityETH	External	Payable	-
	removeLiquidity	External	✓	-

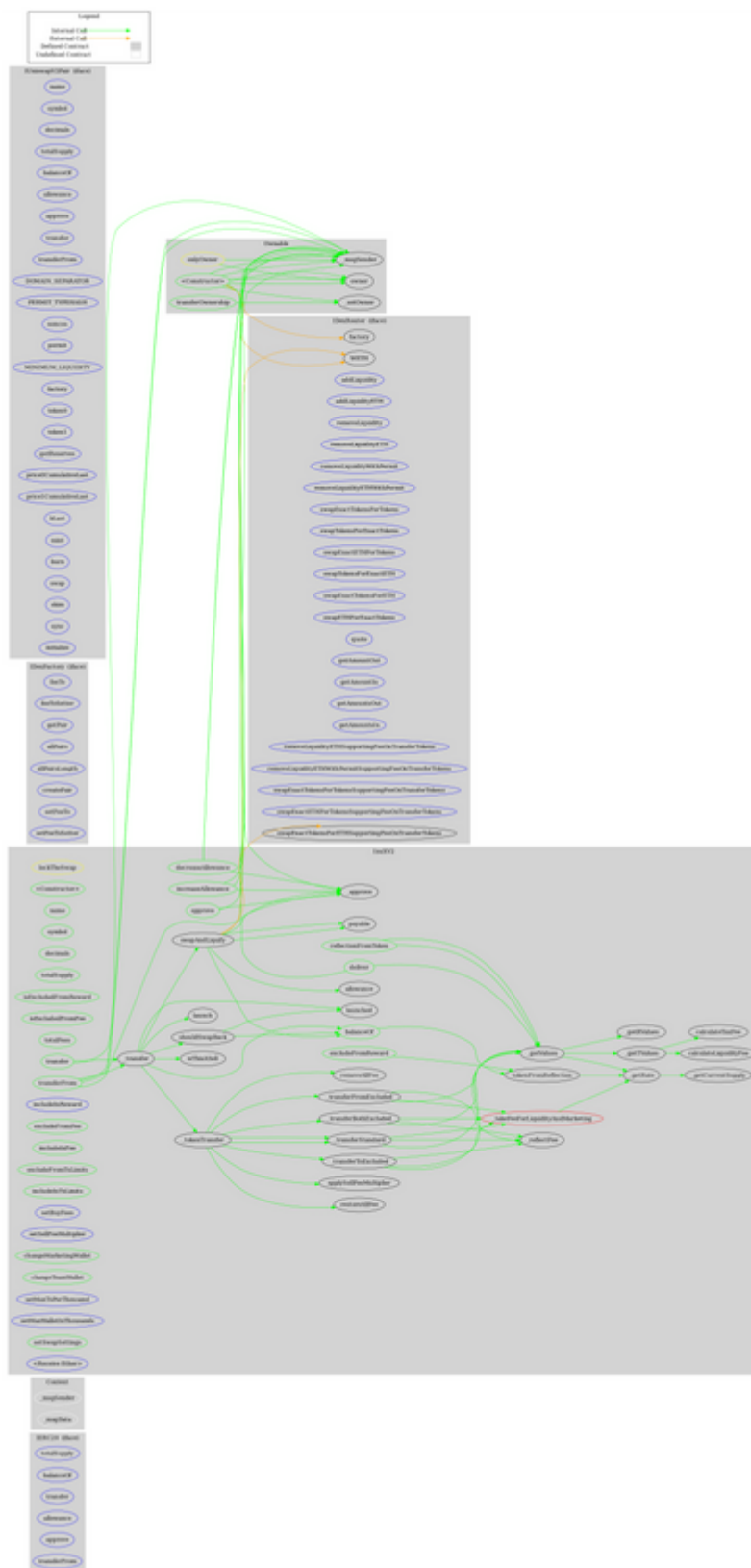
	removeLiquidityETH	External	✓	-
	removeLiquidityWithPermit	External	✓	-
	removeLiquidityETHWithPermit	External	✓	-
	swapExactTokensForTokens	External	✓	-
	swapTokensForExactTokens	External	✓	-
	swapExactETHForTokens	External	Payable	-
	swapTokensForExactETH	External	✓	-
	swapExactTokensForETH	External	✓	-
	swapETHForExactTokens	External	Payable	-
	quote	External		-
	getAmountOut	External		-
	getAmountIn	External		-
	getAmountsOut	External		-
	getAmountsIn	External		-
	removeLiquidityETHSupportingFeeOnTransferTokens	External	✓	-
	removeLiquidityETHWithPermitSupportingFeeOnTransferTokens	External	✓	-
	swapExactTokensForTokensSupportingFeeOnTransferTokens	External	✓	-
	swapExactETHForTokensSupportingFeeOnTransferTokens	External	Payable	-
	swapExactTokensForETHSupportingFeeOnTransferTokens	External	✓	-
<b>InuXV2</b>	Implementation	Context, IERC20, Ownable		
	<Constructor>	Public	✓	-
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	isExcludedFromReward	Public		-
	isExcludedFromFee	Public		-
	totalFees	Public		-
	balanceOf	Public		-
	transfer	Public	✓	-



	allowance	Public		-
	approve	Public	✓	-
	_approve	Private	✓	
	transferFrom	Public	✓	-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	deliver	Public	✓	-
	reflectionFromToken	Public		-
	tokenFromReflection	Public		-
	_getValues	Private		
	_getTValues	Private		
	_getRValues	Private		
	_getRate	Private		
	_getCurrentSupply	Private		
	excludeFromReward	Public	✓	onlyOwner
	includeInReward	External	✓	onlyOwner
	excludeFromFee	Public	✓	onlyOwner
	includeInFee	Public	✓	onlyOwner
	excludeFromTxLimits	Public	✓	onlyOwner
	includeInTxLimits	Public	✓	onlyOwner
	setBuyFees	External	✓	onlyOwner
	setSellFeeMultiplier	External	✓	onlyOwner
	changeMarketingWallet	Public	✓	onlyOwner
	changeTeamWallet	Public	✓	onlyOwner
	setMaxTxPerThousand	External	✓	onlyOwner
	setMaxWalletInThousands	External	✓	onlyOwner
	setSwapSettings	Public	✓	onlyOwner
	_reflectFee	Private	✓	
	calculateTaxFee	Private		
	calculateLiquidityFee	Private		
	isThisASell	Internal	✓	
	_takeFeeForLiquidityAndMarketing	Private	✓	
	removeAllFee	Private	✓	
	restoreAllFee	Private	✓	
	applySellFeeMultiplier	Internal	✓	

	launched	Internal		
	launch	Internal	✓	
	_transfer	Private	✓	
	_tokenTransfer	Private	✓	
	shouldSwapBack	Internal		
	swapAndLiquify	Private	✓	lockTheSwap
	_transferStandard	Private	✓	
	_transferToExcluded	Private	✓	
	_transferFromExcluded	Private	✓	
	_transferBothExcluded	Private	✓	
	<Receive Ether>	External	Payable	-

# Contract Flow



## Domain Info

<b>Domain Name</b>	
<b>Registry Domain ID</b>	2671597431_DOMAIN_COM-VRSN
<b>Creation Date</b>	2022-01-28T19:52:00.00Z
<b>Updated Date</b>	0001-01-01T00:00:00.00Z
<b>Registry Expiry Date</b>	
<b>Registrar WHOIS Server</b>	whois.namecheap.com
<b>Registrar URL</b>	http://www.namecheap.com
<b>Registrar</b>	NAMECHEAP INC
<b>Registrar IANA ID</b>	1068

The domain has been created 6 days before the creation of the audit.

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

## Summary

There are some functions that can be abused by the owner, like manipulating fees and preventing sales. These actions can only happen after the 57600th block number since the token launch. 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>