Cyberscope

Audit Report

Kronos Dao

March 2022

Type	BEP20
Network	BSC
Address	0x30e60C13a7d114344258DEc5822c56ce06c96256
Audited by	© cyberscope

Table of Contents

Contract Review

Contract Name	KronosDaoToken
Compiler Version	v0.7.5+commit.eb77ed08
Optimization	200 runs
Licence	MIT
Explorer	https://bscscan.com/token/0x30e60C13a7d114344258DEc5822c56ce06c96256
Symbol	KRONOS
Decimals	9
Total Supply	2,000,000
Source	contract.sol
Domain	kronosdao.ai

Audit Updates

Initial Audit	23rd March 2022
Corrected	28rd March 2022

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

MT - Mint Tokens

Criticality	critical
Location	contract.sol#L1172

Description

The contract owner has the authority to mint tokens. The owner may take advantage of it by calling the `mint` function. As a result the contract tokens will be highly inflated.

```
    function mint(address account_, uint256 amount_) external onlyVault {
        _mint(account_, amount_);
    }
```

Recommendation

The owner should carefully manage the credentials of the owner's account. We advised considering an extra-strong security mechanism that the actions may be quarantined by many users instead of one. The owner could also renounce the contract ownership for a period of time or pass the access to the zero address.

Contract Diagnostics

🔴 Critical 🟠 Medium ⚪ Minor

Severity	Code	Description
⚪	L01	Public Function could be Declared External
⚪	L04	Conformance to Solidity Naming Conventions
⚪	L05	Unused State Variable
⚪	L06	Missing Events Access Control
⚪	L09	Dead Code Elimination
⚪	L14	Uninitialized Variables in Local Scope

L01 - Public Function could be Declared External

Criticality	minor
Location	contract.sol#L802,806,810,814,824,844,854,871,884,1022 and 7 more

Description

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

```
burnFrom
burn
vault
transferOwnership
renounceOwnership
owner
nonces
permit
decreaseAllowance
...
```

Recommendation

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

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contract.sol#L771,774,777,780,783,786,1001,1072,1114,1156 and 1 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.

```
_burnFrom
_enable
_vault
_owner
DOMAIN_SEPARATOR
_decimals
_symbol
_name
_totalSupply
...
```

Recommendation

Follow the Solidity naming convention.
https://docs.soliditylang.org/en/v0.4.25/style-guide.html#naming-conventions

L05 - Unused State Variable

Criticality	minor
Location	contract.sol#L767

Description

There are segments that contain unused state variables.

```
ERC20TOKEN_ERC1820_INTERFACE_ID
```

Recommendation

Remove unused state variables.

L06 - Missing Events Access Control

Criticality	minor
Location	contract.sol#L1116

Description

Detected missing events for critical access control parameters. There are functions that have no event emitted, so it is difficult to track off-chain changes.

```
_vault = vault_
```

Recommendation

Emit an event for critical parameter changes.

L09 - Dead Code Elimination

Criticality	minor
Location	contract.sol#L973,27,108,83,120,134,94,45,387,294 and 31 more

Description

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

```
substractPercentage
sqrrt
quadraticPricing
percentageOfTotal
percentageAmount
bondingCurve
average
remove
length
...
```

Recommendation

Remove unused functions.

L14 - Uninitialized Variables in Local Scope

Criticality	minor
Location	contract.sol#L307,404,217

Description

The are variables that are defined in the local scope and are not initialized.

```
bytes4Array_
addressArray
```

Recommendation

All the local scoped variables should be initialized.

Contract Functions

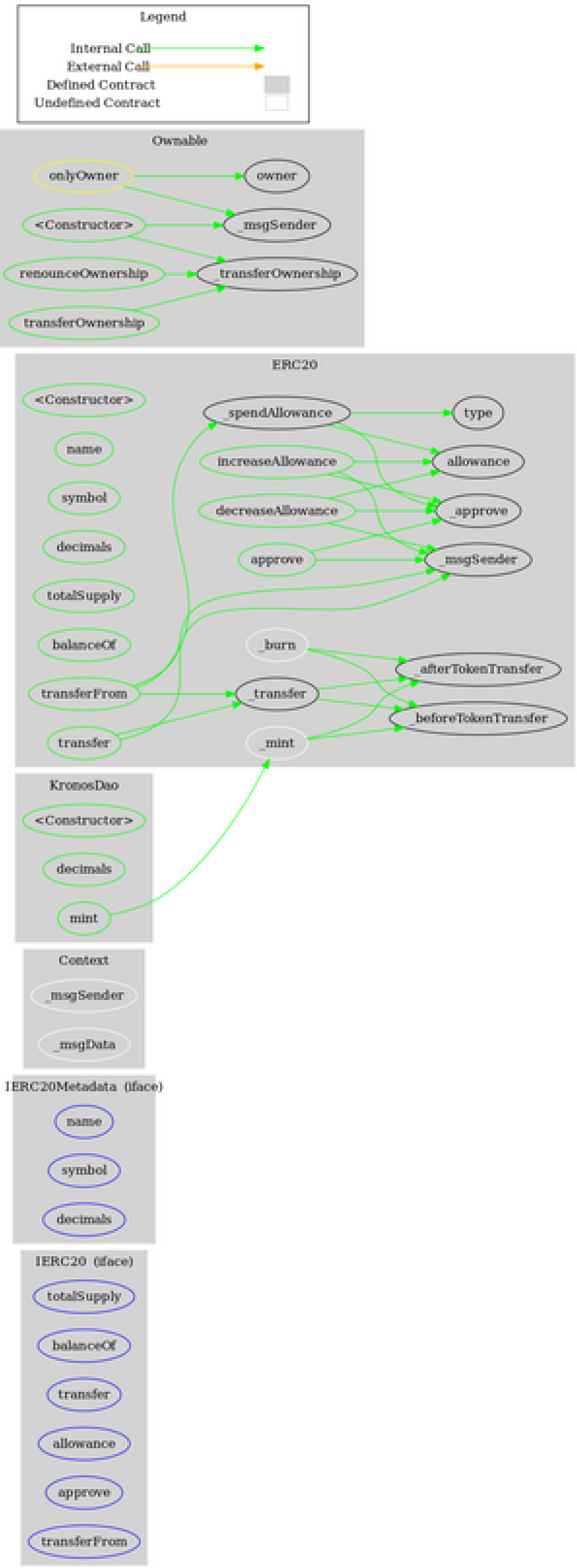
Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
EnumerableSet	Library			
	_add	Private	✓	
	_remove	Private	✓	
	_contains	Private		
	_length	Private		
	_at	Private		
	_getValues	Private		
	_insert	Private	✓	
	add	Internal	✓	
	remove	Internal	✓	
	contains	Internal		
	length	Internal		
	at	Internal		
	getValues	Internal		
	insert	Internal	✓	
	add	Internal	✓	
	remove	Internal	✓	
	contains	Internal		
	length	Internal		
	at	Internal		
	getValues	Internal		
	insert	Internal	✓	
	add	Internal	✓	
	remove	Internal	✓	
	contains	Internal		
	length	Internal		
	at	Internal		
	getValues	Internal		

	insert	Internal	✓	
	add	Internal	✓	
	remove	Internal	✓	
	contains	Internal		
	length	Internal		
	at	Internal		
	add	Internal	✓	
	remove	Internal	✓	
	contains	Internal		
	length	Internal		
	at	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		
	sqrrt	Internal		
	percentageAmount	Internal		
	substractPercentage	Internal		
	percentageOfTotal	Internal		
	average	Internal		
	quadraticPricing	Internal		

	bondingCurve	Internal		
ERC20	Implementation	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	✓	
	_beforeTokenTransfer	Internal	✓	
Counters	Library			
	current	Internal		
	increment	Internal	✓	
	decrement	Internal	✓	
IERC2612Permit	Interface			
	permit	External	✓	-
	nonces	External		-
ERC20Permit	Implementation	ERC20, IERC2612Permit		
	<Constructor>	Public	✓	-
	permit	Public	✓	-

	nonces	Public		-
IOwnable	Interface			
	owner	External		-
	renounceOwnership	External	✓	-
	transferOwnership	External	✓	-
Ownable	Implementation	IOwnable		
	<Constructor>	Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
VaultOwned	Implementation	Ownable		
	setVault	External	✓	onlyOwner
	vault	Public		-
IPinkAntiBot	Interface			
	setTokenOwner	External	✓	-
	onPreTransferCheck	External	✓	-
KronosDaoToken	Implementation	ERC20Permit, VaultOwned		
	<Constructor>	Public	✓	ERC20
	mint	External	✓	onlyVault
	setEnableAntiBot	External	✓	onlyOwner
	burn	Public	✓	-
	burnFrom	Public	✓	-
	_burnFrom	Public	✓	-
	_transfer	Internal	✓	

Contract Flow



Domain Info

Domain Name	kronosdao.ai
Registry Domain ID	1383733_nic_ai
Creation Date	2022-01-20T15:15:24.983Z
Updated Date	-
Registry Expiry Date	-
Registrar WHOIS Server	whois.nic.ai
Registrar URL	-
Registrar	Namecheap
Registrar IANA ID	-

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

Summary

The Smart Contract analysis reported one critical issue. The contract Owner has the ability to mint tokens. As a result the contract tokens will be highly inflated. Apart from that, the contract owner can access some functions that can not be used in a malicious way to disturb the users' transactions. 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