Cyberscope

Audit Report

ASKJA

May 2022

Github	https://github.com/gillesrusticity/askjaCoin
Commit	934f703475d81f006f21e62981485c5e694e1ae9
Audited by	© cyberscope

Table of Contents

Contract Review

Contract Name	AskjaCoin
Github	https://github.com/gillesrusticity/askjaCoin
Commit	934f703475d81f006f21e62981485c5e694e1ae9
Testing Deploy	https://bscscan.com/token/0xF5ACe1eAE0425C5AC6693E22c633F91aEb2533C9
Domain	askja.io

Audit Updates

Initial Audit	9th May 2022
Corrected	

Source Files

Filename	SHA256
@openzeppelin/contracts-upgradeable/access/OwnableUpgradeable.sol	f0cbb88e6cbc994b565645eabd4320d27d529c7f1f4b3abb5fc263f3961c0a24
@openzeppelin/contracts-upgradeable/proxy/utils/Initializable.sol	6e058aaee8c641107b209b62c34d484f2f125a44ecb66f7204a701614dfc1d68
@openzeppelin/contracts-upgradeable/token/ERC20/ERC20Upgradeable.sol	a439a162881f7f36131b1fe307aa2a8dc98ac3f01ac121ff92fbbc25d0d216b5
@openzeppelin/contracts-upgradeable/token/ERC20/extensions/ERC20BurnableUpgradeable.sol	ca660e828b0c4be205a9f56f3b87b91c1fa67cfd0f6e9dbd431faea7a6280d36
@openzeppelin/contracts-upgradeable/token/ERC20/extensions/IERC20MetadataUpgradeable.sol	68bcca423fc72ec9625e219c9e36306c726a347e43f3711467c579bd3f6500c8
@openzeppelin/contracts-upgradeable/token/ERC20/IERC20Upgradeable.sol	db1d80b38061ba675444e6ad861a621d99666042950278d6cdeae9a108afdd17

@openzeppelin/contracts-upgradeable/token/ERC20/presets/ERC20PresetFixedSupplyUpgradeable.sol	06b9cebc2c9e8b7e6a1a4313018d7b9b17255996060211ec73c7b4c053b71123
@openzeppelin/contracts-upgradeable/utils/AddressUpgradeable.sol	44edc4d7099c781d11421cea2d82a52948e738f5f6191c8ad01dfc0f9858549c
@openzeppelin/contracts-upgradeable/utils/ContextUpgradeable.sol	5fb301961e45cb482fe4e05646d2f529aa449fe0e90c6671475d6a32356fa2d4
@openzeppelin/contracts/utils/Strings.sol	8597c62818dcbc6cf85c21179b90b714fb4f70a4347ca2eed23e88c87b08b8a1
contracts/AskjaCoin.sol	67ac86b216c9e2815a730e1c25e6b19789dc3a225978d518bcff3dc9ecb41db6

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

ST - Stop Transactions

Criticality	critical
Location	contract.sol#L58

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 `cooldownPeriod` to a high value. As a result, the users will be able to buy but not be able to sell later.

```
require(
    _lastTimeMinted[_msgSender()] + cooldownPeriod <=
        block.timestamp,
    "not allowed to mint under cooldown period"
);
```

Recommendation

The contract could embody a check for not allowing setting the cooldownPeriod 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.

Contract Diagnostics

Critical Medium Minor

Severity	Code	Description
	DSM	Data Structure Misuse
	L01	Public Function could be Declared External
	L04	Conformance to Solidity Naming Conventions

DSM - Data Structure Misuse

Criticality	minor
Location	contract.sol#L126

Description

The contract uses the valuable _whiteList as an array. The business logic of the contract does not require to iterate this structure sequentially. Thus, unnecessary loops are produced that increase the required gas.

```
function isWhiteListedAddress(address _address) public view returns (bool) {
    bool status = false;
    for (uint256 i = 0; i < _whiteList.length; i++) {
        if (_whiteList[i] == _address) {
            return true;
        }
    }
    return status;
}
```

Recommendation

The contract could use a data structure that provides instant access. For instance, a Set or a Map would fit better to the business logic of the contract. This way the time complexity will be reduced from o(n) to o(1).

L01 - Public Function could be Declared External

Criticality	minor
Location	contracts/AskjaCoin.sol#L49,91,100,104,116,121,137,141

Description

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

```
getFeeDisabled
updateFeeDisabled
addToWhiteList
getWhiteList
setCoolDownValue
getCoolDownValue
getCoolDownAddress
transfer
```

Recommendation

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

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contracts/AskjaCoin.sol#L25,104,121,126,13,20,22,30

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.

```
_feeDisabled
_whiteList
_lastTimeMinted
rusticityFee
_address
_index
_cooldownPeriod
coolDownChanged
```

Recommendation

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

Contract Functions

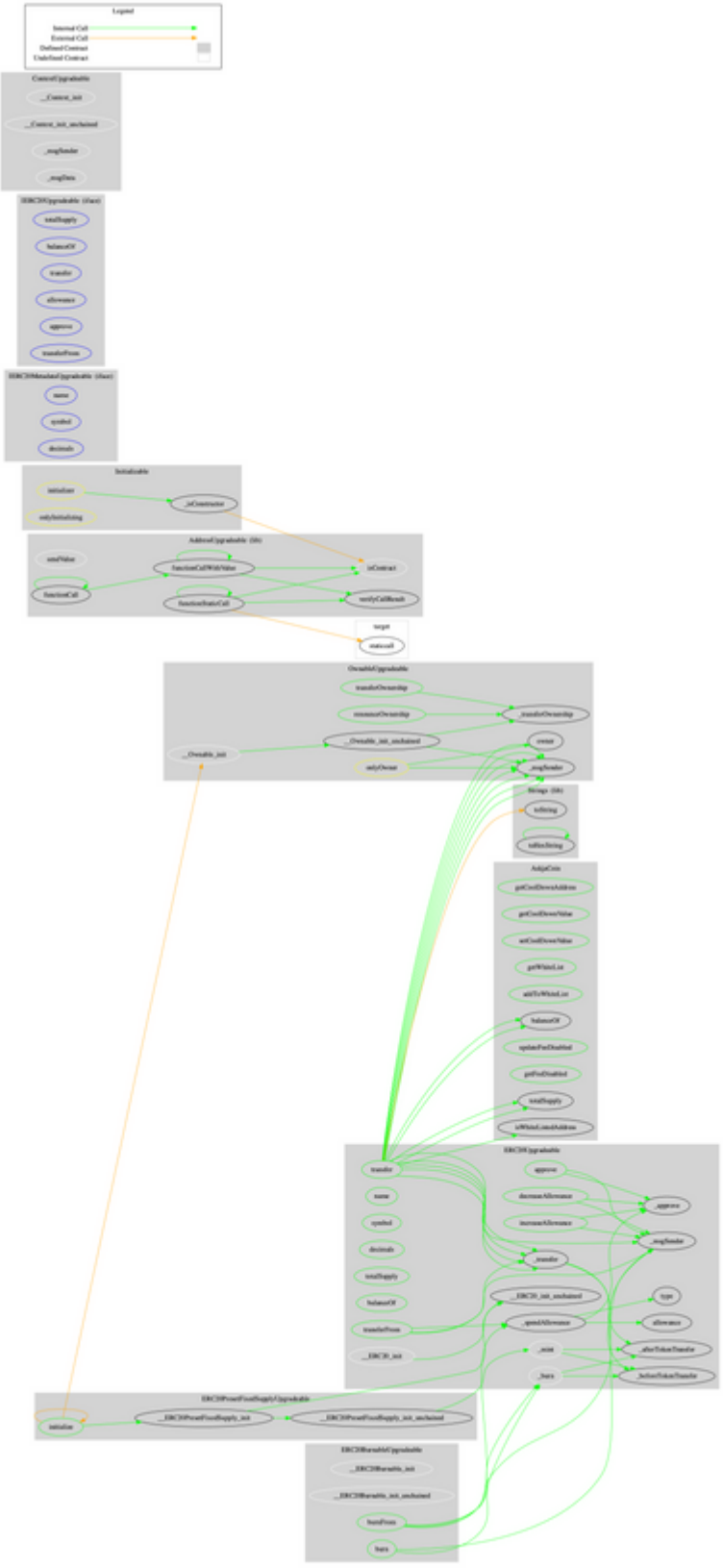
Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
OwnableUpgradeable	Implementation	Initializable, ContextUpgradeable		
	__Ownable_init	Internal	✓	onlyInitializing
	__Ownable_init_unchained	Internal	✓	onlyInitializing
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
	_transferOwnership	Internal	✓	
Initializable	Implementation			
	_isConstructor	Private		
ERC20Upgradeable	Implementation	Initializable, ContextUpgradeable, IERC20Upgradeable, IERC20MetadataUpgradeable		
	__ERC20_init	Internal	✓	onlyInitializing
	__ERC20_init_unchained	Internal	✓	onlyInitializing
	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	✓	
	_spendAllowance	Internal	✓	
	_beforeTokenTransfer	Internal	✓	
	_afterTokenTransfer	Internal	✓	
ERC20BurnableUpgradeable	Implementation	Initializable, ContextUpgradeable, ERC20Upgradeable		
	__ERC20Burnable_init	Internal	✓	onlyInitializing
	__ERC20Burnable_init_unchained	Internal	✓	onlyInitializing
	burn	Public	✓	-
	burnFrom	Public	✓	-
IERC20MetadataUpgradeable	Interface	IERC20Upgradeable		
	name	External		-
	symbol	External		-
	decimals	External		-
IERC20Upgradeable	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
ERC20PresetFixedSupplyUpgradeable	Implementation	Initializable, ERC20BurnableUpgrad		

		eable		
	initialize	Public	✓	initializer
	__ERC20PresetFixedSupply_init	Internal	✓	onlyInitializing
	__ERC20PresetFixedSupply_init_unchained	Internal	✓	onlyInitializing
AddressUpgradeable	Library			
	isContract	Internal		
	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionStaticCall	Internal		
	functionStaticCall	Internal		
	verifyCallResult	Internal		
ContextUpgradeable	Implementation	Initializable		
	__Context_init	Internal	✓	onlyInitializing
	__Context_init_unchained	Internal	✓	onlyInitializing
	_msgSender	Internal		
	_msgData	Internal		
Strings	Library			
	toString	Internal		
	toHexString	Internal		
	toHexString	Internal		
AskjaCoin	Implementation	ERC20PresetFixedSupplyUpgradeable, OwnableUpgradeable		
	initialize	Public	✓	initializer
	transfer	Public	✓	-

	getCoolDownAddress	Public		-
	getCoolDownValue	Public		-
	setCoolDownValue	Public	✓	onlyOwner
	getWhiteList	Public		-
	addToWhiteList	Public	✓	onlyOwner
	isWhiteListedAddress	Public		-
	updateFeeDisabled	Public	✓	onlyOwner
	getFeeDisabled	Public		-

Contract Flow



Domain Info

Domain Name	askja.io
Registry Domain ID	6712323d52e14f7b9180b158889988b5-DONUTS
Creation Date	2022-03-28T10:58:50Z
Updated Date	2022-04-02T10:59:15Z
Registry Expiry Date	2023-03-28T10:58:50Z
Registrar WHOIS Server	whois.godaddy.com/
Registrar URL	http://www.godaddy.com/domains/search.aspx?ci=8990
Registrar	GoDaddy.com, LLC
Registrar IANA ID	146

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

The Smart Contract analysis reported one critical severity issue. The contract owner has the authority to stop transactions. The contract can be converted into a honeypot and prevent users from selling if the owner abuses the admin functions. A multi-wallet signing pattern will provide security against potential hacks. Temporarily locking the contract or renouncing ownership will eliminate all the contract threats. The fee is fixed to 7%.

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