



RUGFREECOINS



Drip X Token

RugfreeCoins Verified on March 07th, 2024

Overview

- ✗ Mint function found, the owner can mint tokens after initial deployment.
- ✓ The owner can't set a max transaction limit
- ✓ The owner can't pause trading once it's enabled
- ✓ The owner can't change fees.
- ✗ The owner can blacklist wallets.
- ✓ The owner can't set a max wallet limit
- ✗ The owner can claim the contract's balance of its own token.

! HIGH SEVERITY ISSUES

The contract is an upgradable contract, the owner can change the functions later

The owner can mint new tokens without a limit

```
function mint(address _to, uint256 _amount) public onlyManager {  
    balances[_to] += _amount;  
    totalSupply += _amount;  
    emit Transfer(address(0), _to, _amount);  
}
```

The owner can burn tokens from any wallet without their permissions

```
function burnFrom(address _from, uint256 _amount) public onlyManager {  
    _burn(_from, _amount);  
}
```

The owner can blacklist wallets

```
function setIsBlacklisted(address user, bool value) public onlyOwner {  
    isBlacklisted[user] = value;  
}
```

The owner can withdraw native tokens from the contract

```
function withdrawTokens() public onlyOwner {  
    _transfer(address(this), owner(), balanceOf(address(this)));  
}
```

Contents

Overview.....	2
Contents.....	4
Audit details.....	5
Disclaimer.....	6
Background.....	7
Tokenomics.....	8
Target market and the concept.....	9
Potential to grow with score points.....	10
Contract details.....	11
Contract code function details.....	12
Contract description table.....	13
Inheritance Hierarchy.....	16
Security issue checking status.....	17
Owner privileges.....	19
Audit conclusion.....	21

Audit details



Audited project

DripX Token



Contract Address

0xeCbcE2c13d3A1248deBfD23DCc0558b495916756



Client contact

DripX Token Team



Blockchain

Binance Smart chain



Project website

<https://www.dripX.win/>

Disclaimer

This is a limited report on our findings based on our analysis, in accordance with good industry practice as at the date of this report, in relation to cybersecurity vulnerabilities and issues in the framework and algorithms based on smart contracts, the details of which are set out in this report. In order to get a full view of our analysis, it is crucial for you to read the full report. While we have done our best in conducting our analysis and producing this report, it is important to note that you should not rely on this report and cannot claim against us on the basis of what it says or doesn't say, or how we produced it, and it is important for you to conduct your own independent investigations before making any decisions. We go into more detail on this in the disclaimer below – **please make sure to read it in full.**

❗ DISCLAIMER

By reading this report or any part of it, you agree to the terms of this disclaimer. If you do not agree to the terms, then please immediately cease reading this report, and delete and destroy any and all copies of this report downloaded and/or printed by you. **This report is provided for information purposes only and on a non-reliance basis, and does not constitute investment advice.** No one shall have any right to rely on the report or its contents, and **RugfreeCoins and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (RugfreeCoins) owe no duty of care towards you or any other person**, nor does RugfreeCoins make any warranty or representation to any person on the accuracy or completeness of the report.

The report is provided "as is", without any conditions, warranties or other terms of any kind except as set out in this disclaimer, and RugfreeCoins hereby excludes all representations, warranties, conditions and other terms (including, without limitation, the warranties implied by law of satisfactory quality, fitness for purpose and the use of reasonable care and skill) which, but for this clause, might have effect in relation to the report. Except and only to the extent that it is prohibited by law, RugfreeCoins hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against RugfreeCoins, for any amount or kind of loss or damage that may result to you or any other person (including without limitation, any direct, indirect, special, punitive, consequential or pure economic loss or damages, or any loss of income, profits, goodwill, data, contracts, use of money, or business interruption, and whether in delict, tort (including without limitation negligence), contract, breach of statutory duty, misrepresentation (whether innocent or negligent) or otherwise under any claim of any nature whatsoever in any jurisdiction) in any way arising from or connected with this report and the use, inability to use or the results of use of this report, and any reliance on this report.

The analysis of the security is purely based on the smart contracts alone. No applications or operations were reviewed for security. No product code has been reviewed.

Background

RugfreeCoins was commissioned by the **DripX Token Team** to perform an audit of the smart contract.

<https://bscscan.com/address/0xCbcE2c13d3A1248deBfD23DCc0558b495916756>

This audit focuses on verifying that the smart contract is secure, resilient, and working according to the specifications.

The information in this report should be used to understand the risk exposure of the smart contract, project feasibility, and long-term sustainability, and as a guide to improving the smart contract's security posture by remediating the identified issues.












Tokenomics

▲ 0% tax when buying & selling

Target market and the concept

- ▶ Anyone who's interested in the Crypto space with long-term investment plans.
- ▶ Anyone who's ready to earn a passive income by holding tokens.
- ▶ Anyone who's interested in trading tokens.
- ▶ Anyone who's interested in taking part in the DripX token ecosystem.
- ▶ Anyone who's interested in taking part in the future plans of DripX Token.
- ▶ Anyone who's interested in making financial transactions with any other party DripX Token as the currency.

Potential to grow with score points

 Project efficiency	9 / 10
 Project uniqueness	9 / 10
 Information quality	9 / 10
 Service quality	9 / 10
 System quality	9 / 10
 Impact on the community	9 / 10
 Impact on the business	9 / 10
 Preparing for the future	9 / 10
 Smart contract security	7 / 10
 Smart contract functionality assessment	10 / 10
 Total Score	8.9 / 10

Contract details

Token contract details for 07th of March 2024









Contract name	DripX Token
Contract address	0xeCbcE2c13d3A1248deBfD23DCc0558b495916756
Token supply	336,369,294,645.906426
Token ticker	DRIPX
Decimals	18
Token holders	4175
Transaction count	63379
Contract deployer address	0xe90C5C1D36aB80FfcCCca40C4989633026EF45Fa



















Contract code function details

Nº	Category	Item	Result
1	Coding conventions	ERC20 Token standards	PASS ▾
		Compile errors	PASS ▾
		Compiler version security	PASS ▾
		Visibility specifiers	PASS ▾
		Gas consumption	PASS ▾
		SafeMath features	PASS ▾
		Fallback usage	PASS ▾
		tx.origin usage	PASS ▾
		Deprecated items	PASS ▾
		Redundant code	PASS ▾
2	Function call audit	Overriding variables	PASS ▾
		Authorization of function call	PASS ▾
		Low level function (call/delegate call) security	PASS ▾
		Returned value security	PASS ▾
		Self destruct function security	PASS ▾
3	Business security & centralisation	Access control of owners	HIGH ▾
		Business logics	PASS ▾
		Business implementation	PASS ▾
4	Integer overflow/underflow		PASS ▾
5	Reentrancy		PASS ▾
6	Exceptional reachable state		PASS ▾
7	Transaction ordering dependence		PASS ▾
8	Block properties dependence		PASS ▾
9	Pseudo random number generator (PRNG)		PASS ▾
10	DoS (Denial of Service)		PASS ▾
11	Token vesting implementation		PASS ▾
12	Fake deposit		PASS ▾
13	Event security		PASS ▾

Contract description table

The below table represents the summary of the contracts and methods in the token contract. We scanned the whole contract and listed down all the Interfaces, functions, and implementations with their visibility and mutability.

Contract	Type	Bases		
L	Function Name	Visibility	Mutability	Modifiers
Admin Upgradeability Proxy	Implementation	TransparentUpgradeableProxy		
L		Public !		Transparent Upgradeable Proxy
TransparentUpgradeableProxy	Implementation	ERC1967Proxy		
L		Public !		ERC1967Proxy
L	admin	External !		ifAdmin
L	implementation	External !		ifAdmin
L	changeAdmin	External !		ifAdmin
L	upgradeTo	External !		ifAdmin
L	upgradeToAndCall	External !		ifAdmin
L	_admin	Internal 		
L	_beforeFallback	Internal 		
BeaconProxy	Implementation	Proxy, ERC1967 Upgrade		
L		Public !		NO !
L	_beacon	Internal 		
L	_implementation	Internal 		
L	_setBeacon	Internal 		

Upgradeable Beacon	Implementation	IBeacon, Ownable		
L		Public !		NO !
L	implementation	Public !		NO !
L	upgradeTo	Public !		onlyOwner
L	_setImplementation	Private 		
ERC1967Proxy	Implementation	Proxy, ERC1967 Upgrade		
L		Public !		NO !
L	_implementation	Internal 		
ProxyAdmin	Implementation	Ownable		
L	getProxyImplementation	Public !		NO !
L	getProxyAdmin	Public !		NO !
L	changeProxyAdmin	Public !		onlyOwner
L	upgrade	Public !		onlyOwner
L	upgradeAndCall	Public !		onlyOwner
IBeacon	Interface			
L	implementation	External !		NO !
Proxy	Implementation			
L	_delegate	Internal 		
L	_implementation	Internal 		
L	_fallback	Internal 		
L		External !		NO !
L		External !		NO !
L	_beforeFallback	Internal 		

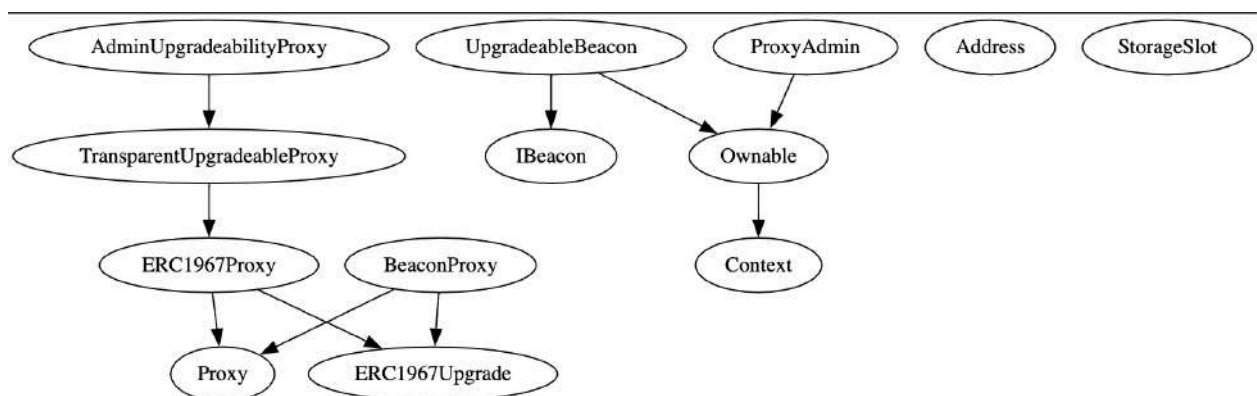
ERC1967 Upgrade	Implementation			
L	_getImplementation	Internal 🔒		
L	_setImplementation	Private 🔑	🛑	
L	_upgradeTo	Internal 🔒	🛑	
L	_upgradeToAndCall	Internal 🔒	🛑	
L	_upgradeToAndCallSecure	Internal 🔒	🛑	
L	_upgradeBeaconToAndCall	Internal 🔒	🛑	
L	_getAdmin	Internal 🔒		
L	_setAdmin	Private 🔑	🛑	
L	_changeAdmin	Internal 🔒	🛑	
L	_getBeacon	Internal 🔒		
L	_setBeacon	Private 🔑	🛑	
Address	Library			
L	isContract	Internal 🔒		
L	sendValue	Internal 🔒	🛑	
L	functionCall	Internal 🔒	🛑	
L	functionCall	Internal 🔒	🛑	
L	functionCallWithValue	Internal 🔒	🛑	
L	functionCallWithValue	Internal 🔒	🛑	
L	functionStaticCall	Internal 🔒		
L	functionStaticCall	Internal 🔒		
L	functionDelegateCall	Internal 🔒	🛑	
L	functionDelegateCall	Internal 🔒	🛑	
L	_verifyCallResult	Private 🔑		
StorageSlot	Library			
L	getAddressSlot	Internal 🔒		
L	getBooleanSlot	Internal 🔒		

L	getBytes32Slot	Internal 🔒		
L	getUint256Slot	Internal 🔒		
Ownable	Implementation	Context		
L		Public !	🛑	NO !
L	owner	Public !		NO !
L	renounceOwnership	Public !	🛑	onlyOwner
L	transferOwnership	Public !	🛑	onlyOwner
Context	Implementation			
L	_msgSender	Internal 🔒		
L	_msgData	Internal 🔒		

Legend

Symbol	Meaning
🛑	Function can modify state
💰	Function is payable

Inheritance Hierarchy



Security issue checking status

❖ High severity issues

The contract is an upgradable contract, the owner can change the functions later

The owner can mint new tokens without a limit

```
function mint(address _to, uint256 _amount) public onlyManager {  
    balances[_to] += _amount;  
    totalSupply += _amount;  
    emit Transfer(address(0), _to, _amount);  
}
```

The owner can burn tokens from any wallet without their permissions

```
function burnFrom(address _from, uint256 _amount) public onlyManager {  
    _burn(_from, _amount);  
}
```

The owner can blacklist wallets

```
function setIsBlacklisted(address user, bool value) public onlyOwner {  
    isBlacklisted[user] = value;  
}
```

The owner can withdraw native tokens from the contract

```
function withdrawTokens() public onlyOwner {  
    _transfer(address(this), owner(), balanceOf(address(this)));  
}
```

❖ Medium severity issues

No medium severity issues found

❖ Low severity issues

No high severity issues found

Owner privileges

- ❖ Owners can mint new tokens without a maximum limit

```
function mint(address _to, uint256 _amount) public onlyManager {  
    balances[_to] += _amount;  
    totalSupply += _amount;  
    emit Transfer(address(0), _to, _amount);  
}
```

- ❖ Owners can burn tokens from any wallet

```
function burnFrom(address _from, uint256 _amount) public onlyManager {  
    _burn(_from, _amount);  
}
```

- ❖ Owner can blacklist/whitelist wallets

```
function setIsBlacklisted(address user, bool value) public onlyOwner {  
    isBlacklisted[user] = value;  
}
```

- ❖ Owner can withdraw native tokens from the contract

```
function withdrawTokens() public onlyOwner {  
    _transfer(address(this), owner(), balanceOf(address(this)));  
}
```

Audit conclusion

RugFreeCoins team has performed in-depth testing, line-by-line manual code review, and automated audit of the smart contract. The smart contract was analyzed mainly for common smart contract vulnerabilities, exploits, manipulations, and hacks. According to the smart contract audit.

Smart contract functional Status:	PASS ▾
Smart contract security Status:	HIGH ISSUES ▾
Number of risk issues:	04
Solidity code functional issue level:	PASS ▾
Number of owner privileges:	04
Centralization risk correlated to the active owner:	HIGH ▾
Smart contract active ownership:	ACTIVE ▾