



Smart Contract Security Audit

Audit details:

Audited project:	Zombies Farm
Deployer address	0xba976f790c86359ff8233b3581c89176a6dddd7a
Blockchain:	Binance Smart Chain
Project website:	Not provided

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 below 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 TechRate and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (TechRate) owe no duty of care towards you or any other person, nor does TechRate 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 TechRate 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, TechRate hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against TechRate, 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

TechRate was commissioned by Zombies Farm to perform an audit of smart contracts:

- <https://bscscan.com/address/0x5DCC4648d7029C7055fcF25bED6CfFc99E23727E#code>

The purpose of the audit was to achieve the following:

- Ensure that the smart contract functions as intended.
- Identify potential security issues with the smart contract.

The information in this report should be used to understand the risk exposure of the smart contract, and as a guide to improve the security posture of the smart contract by remediating the issues that were identified.

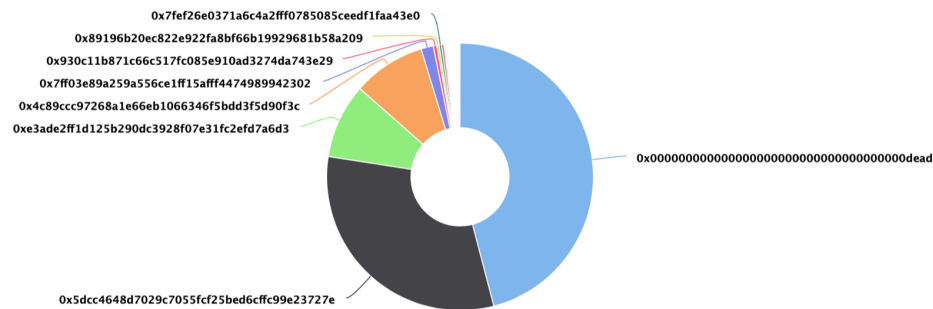
Zombies Farm token distribution

💡 The top 100 holders collectively own 99.89% (265,075.27 Tokens) of Zombies Farm

💡 Token Total Supply: 265,370.69 Token | Total Token Holders: 405




Zombies Farm Top 100 Token Holders

Source: BscScan.com



(A total of 265,075.27 tokens held by the top 100 accounts from the total supply of 265,370.69 token)

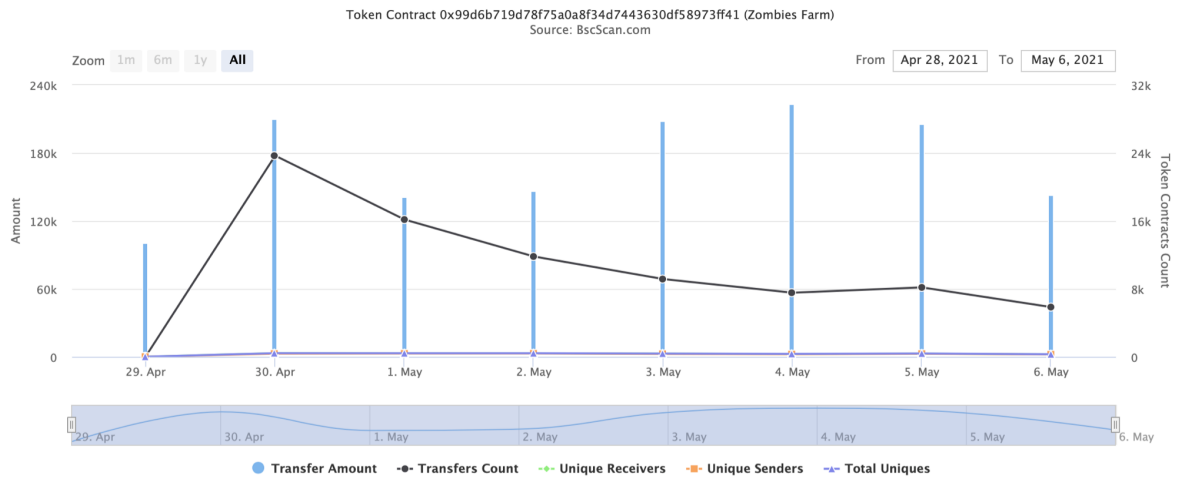
Zombies Farm top 10 token holders

Rank	Address	Quantity (Token)	Percentage
1	0x00000000000000000000000000000000dead	121,850.833237696171107236	45.9172%
2	 0x5dcc4648d7029c7055fc25bed6ffc99e23727e	83,673.284198380980501807	31.5307%
3	 0xe3ade2ff1d125b290dc3928f07e31fc2efd7a6d3	23,968.431582655065982309	9.0321%
4	 0x4c89ccc97268a1e6eb1066346f5bdd3f5d90f3c	23,407.771524877929632851	8.8208%
5	0x7f03e89a259a556ce1ff15afff4474989942302	3,918.61999999999999999078	1.4767%
6	0x930c11b871c66c517fc085e910ad33274da743e29	1,133.0411555555555555164	0.4270%
7	0x89196b20ec822e922fa8bf66b19929681b58a209	787.868968687512626806	0.2969%
8	0x7fef26e0371a6c4a2fff0785085ceedf1faa43e0	763.606909332028486616	0.2878%
9	0x64c5e9b889f72e74db31b599e55cd8584276c744	733.493962912787034759	0.2764%
10	0xe641b801776e668356717134ed268afcb9b188c	491.881421292725261688	0.1854%

Zombies Farm transactions

Time Series: Token Contract Overview

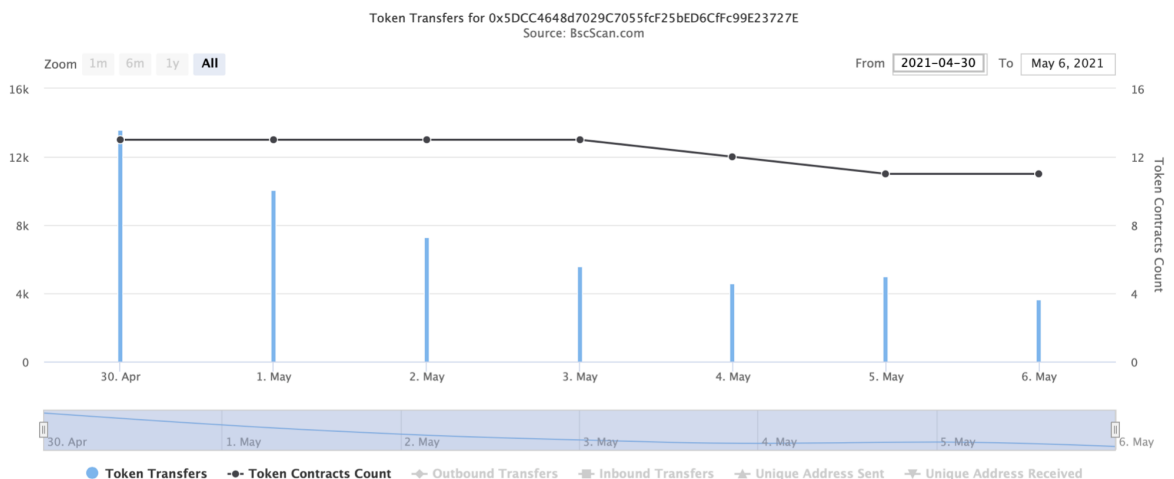
Thu 29, Apr 2021 - Thu 6, May 2021



Masterchef transactions

Time Series: Address Token (BEP-20) Transfers

Fri 30, Apr 2021 - Thu 6, May 2021



Pro-Tip: Click on the chart data points to view more

MasterChef contract details for 07.05.2021.

Contract name:	MasterChef
Compiler version:	v0.6.12+commit.27d51765
Contract address:	0x5DCC4648d7029C7055fcF25bED6CfFc99E23727E
Deployer address:	0xba976f790c86359ff8233b3581c89176a6dddd7a
Dev address:	0x7ff03e89a259a556ce1ff15afff4474989942302
Fee address:	0x71a93a1efce449673e5938e692f23c96d95180b6
Zombie contract address:	0x99d6b719d78f75a0a8f34d7443630df58973ff41
Zombie per block:	700000000000000000
Contract owner address:	0x492196c8bb21d560f7092e69532da7954631655e
Pool length:	13
Start block:	7001777
Total alloc point:	3150
Bonus multiplier:	1

MasterChef functions outline

+ [Lib] Address

- [Int] isContract
- [Int] sendValue #
- [Int] functionCall #
- [Int] functionCall #
- [Int] functionCallWithValue #
- [Int] functionCallWithValue #
- [Prv] _functionCallWithValue #

+ BEP20 (Context, IBEP20, Ownable)

- [Pub] <Constructor> #
- [Ext] getOwner
- [Pub] name
- [Pub] decimals
- [Pub] symbol
- [Pub] totalSupply
- [Pub] balanceOf
- [Pub] transfer #
- [Pub] allowance
- [Pub] approve #
- [Pub] transferFrom #
- [Pub] increaseAllowance #
- [Pub] decreaseAllowance #
- [Pub] mint #
 - modifiers: onlyOwner
- [Int] _transfer #
- [Int] _mint #
- [Int] _burn #
- [Int] _approve #
- [Int] _burnFrom #

+ Context

- [Int] _msgSender
- [Int] _msgData

+ [Int] IBEP20

- [Ext] totalSupply
- [Ext] decimals
- [Ext] symbol
- [Ext] name
- [Ext] getOwner
- [Ext] balanceOf
- [Ext] transfer #
- [Ext] allowance

- [Ext] approve #
- [Ext] transferFrom #
- + Ownable (Context)
 - [Pub] <Constructor> #
 - [Pub] owner
 - [Pub] renounceOwnership #
 - modifiers: onlyOwner
 - [Pub] transferOwnership #
 - modifiers: onlyOwner
- + ReentrancyGuard
- + [Lib] SafeBEP20
 - [Int] safeTransfer #
 - [Int] safeTransferFrom #
 - [Int] safeApprove #
 - [Int] safeIncreaseAllowance #
 - [Int] safeDecreaseAllowance #
 - [Prv] _callOptionalReturn #
- + [Lib] SafeMath
 - [Int] tryAdd
 - [Int] trySub
 - [Int] tryMul
 - [Int] tryDiv
 - [Int] tryMod
 - [Int] add
 - [Int] sub
 - [Int] mul
 - [Int] div
 - [Int] mod
 - [Int] sub
 - [Int] div
 - [Int] mod
- + ZombieToken (BEP20)
 - [Pub] mint #
 - modifiers: onlyOwner
 - [Int] _transfer #
 - [Ext] delegates
 - [Ext] delegate #
 - [Ext] delegateBySig #
 - [Ext] getCurrentVotes
 - [Ext] getPriorVotes
 - [Int] _delegate #
 - [Int] _moveDelegates #

- [Int] `_writeCheckpoint` #
- [Int] `safe32`
- [Int] `getChainId`
- + **MasterChef** (Ownable, ReentrancyGuard)
 - [Pub] `<Constructor>` #
 - [Ext] `poolLength`
 - [Pub] `add` #
 - modifiers: `onlyOwner, nonDuplicated`
 - [Pub] `set` #
 - modifiers: `onlyOwner`
 - [Pub] `getMultiplier`
 - [Ext] `pendingZombie`
 - [Pub] `massUpdatePools` #
 - [Pub] `updatePool` #
 - [Pub] `deposit` #
 - modifiers: `nonReentrant`
 - [Pub] `withdraw` #
 - modifiers: `nonReentrant`
 - [Pub] `emergencyWithdraw` #
 - modifiers: `nonReentrant`
 - [Int] `safeZombieTransfer` #
 - [Pub] `dev` #
 - [Pub] `setFeeAddress` #
 - [Pub] `updateEmissionRate` #
 - modifiers: `onlyOwner`

(\$) = payable function

= non-constant function

Issues Checking Status

№	Issue description.	Checking status
1	Compiler errors.	Passed
2	Race conditions and Reentrancy. Cross-function race conditions.	Passed
3	Possible delays in data delivery.	Passed
4	Oracle calls.	Passed
5	Front running.	Passed
6	Timestamp dependence.	Passed
7	Integer Overflow and Underflow.	Passed
8	DoS with Revert.	Passed
9	DoS with block gas limit.	Low issues
10	Methods execution permissions.	Passed
11	Economy model of the contract.	Passed
12	The impact of the exchange rate on the logic.	Passed
13	Private user data leaks.	Passed
14	Malicious Event log.	Passed
15	Scoping and Declarations.	Passed
16	Uninitialized storage pointers.	Passed
17	Arithmetic accuracy.	Passed
18	Design Logic.	Medium issues
19	Cross-function race conditions.	Passed
20	Safe Open Zeppelin contracts implementation and usage.	Passed
21	Fallback function security.	Passed

Security Issues

High Severity Issues

No high severity issues found.

Medium Severity Issues

1. Wrong burning

Issue:

There is sending tokens to the dead address in overridden `_transfer` functions, instead of burning them in token contract.

Recommendation:

There should be a burn instead of sending to the dead address.

Low Severity Issues

1. Block gas limit

Issue:

The `updateEmissionRate` function can fail due to block gas limit if the pool size is too big.

Conclusion

Smart contracts do not contain high severity issues!

Techrate note:

Please check the disclaimer above and note, the audit makes no statements or warranties on business model, investment attractiveness or code sustainability. The report is provided for the only contract mentioned in the report and does not include any other potential contracts deployed by Owner.