Smart Contract Security Audit

Audit details:

Audited project:	Neonic Finance
Deployer address	0x2a28724f7134c7d4ec3f6d98eacd1a20d9cc58fc
Blockchain:	Binance Smart Chain
Project website:	https://neonic.finance

April, 2021
TechRate

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 Neonic Finance to perform an audit of smart contracts:

- https://bscscan.com/address/0x94026f0227cE0c9611e8a228f114F9F19CC3Fa87#code
- https://bscscan.com/address/0x045502ee488806bdf22928b6228bdd162b5056f6
- https://bscscan.com/address/0xa307cbb816aeedb3c645d58d40dde62364c4be1d#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.

Contracts details

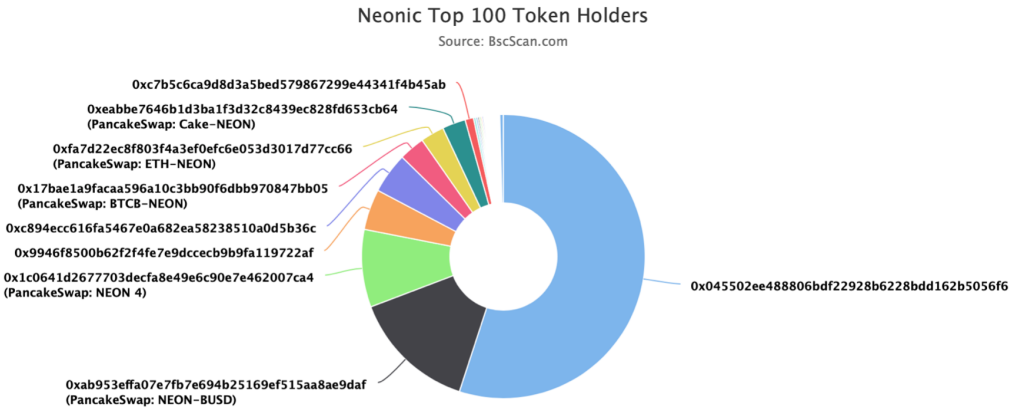
Token contract details for 14.04.2021.

Contract name:	Neonic Finance
Compiler version:	v0.6.12+commit.27d51765
Contract address:	0x94026f0227cE0c9611e8a228f114F9F19CC3Fa87
Total supply:	76_253_912_808_835_167_963_466
Token ticker:	NEON
Decimals:	18
Token holders:	592
Transactions count:	71503
Top 100 holders dominance:	99 %
Contract deployer address:	0x2a28724f7134c7d4ec3f6d98eacd1a20d9cc58fc
Contract's current owner address:	0x045502ee488806bdf22928b6228bdd162b5056f6

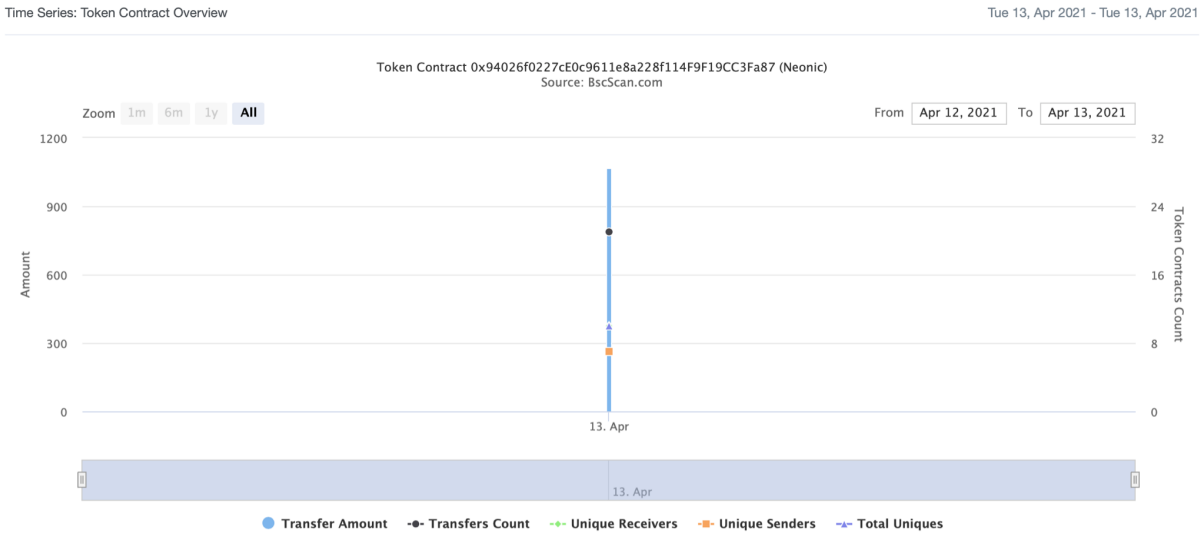
Neonic Finance top 8 token holders

Rank	Address	Quantity (Token)	Percentage
1	0x045502ee488806bdf22928b6228bdd162b5056f6	42,065.835673673953959133	55.0881%
2	PancakeSwap: NEON-BUSD	10,972.647639269972512965	14.3694%
3	PancakeSwap: NEON 4	6,892.803256969393256543	9.0266%
4	0x9946f8500b62f2f4fe7e9dccecb9b9fa119722af	3,533.933616799680220713	4.6279%
5	0xc894ecc616fa5467e0a682ea58238510a0d5b36c	3,533.933616799680220713	4.6279%
6	PancakeSwap: BTCB-NEON	2,270.301496857689319129	2.9731%
7	PancakeSwap: ETH-NEON	2,178.117535707225382094	2.8524%
8	PancakeSwap: Cake-NEON	2,024.609350592058750706	2.6514%

Neonic Finance top 100 token distribution



Neonic Finance contract interaction details



NeonicFactory contract details for 14.04.2021.

Contract name:	NeonicFactory
Compiler version:	v0.6.12+commit.27d51765
Contract address:	0x045502ee488806bdf22928b6228bdd162b5056f6
Dev address:	0x9946f8500b62f2f4fe7e9dccecb9b9fa119722af
Fee address:	0x0fa044d45ae151f25505c1bd11709894ea4739c3
NEON contract address:	0x94026f0227ce0c9611e8a228f114f9f19cc3fa87
NEON per block:	5_000_000_000_000_000_000
Contract owner address:	0xc5722545579ef1b34620dd6298555b9fa3bfbc6b
Pool length:	10
Start block:	6547728
Total alloc point:	644

NeonicFactory contract Pools info:

Pool with id 0:

lpToken *address* : 0x1C0641d2677703DEcfA8E49E6C90E7E462007CA4
allocPoint *uint256* : 120
lastRewardBlock *uint256* : 6558089
accNeonPerShare *uint256* : 176063305807415
depositFeeBP *uint16* : 0

Pool with id 1:

lpToken *address* : 0xaB953EFFA07e7FB7E694b25169ef515Aa8Ae9Daf
allocPoint *uint256* : 200
lastRewardBlock *uint256* : 6558095
accNeonPerShare *uint256* : 9750929653874
depositFeeBP *uint16* : 0

Pool with id 2:

lpToken *address* : 0xEAbBe7646B1D3ba1f3D32c8439ec828fD653cB64
allocPoint *uint256* : 40
lastRewardBlock *uint256* : 6558089
accNeonPerShare *uint256* : 23032068991134
depositFeeBP *uint16* : 0

Pool with id 3:

lpToken *address* : 0xFa7D22ec8F803F4A3eF0efc6e053d3017d77CC66
allocPoint *uint256* : 40
lastRewardBlock *uint256* : 6558090
accNeonPerShare *uint256* : 240421770180988
depositFeeBP *uint16* : 0

Pool with id 4:

lpToken *address* : 0x17baE1a9FaCaA596a10C3BB90F6Dbb970847BB05
allocPoint *uint256* : 40
lastRewardBlock *uint256* : 6558105
accNeonPerShare *uint256* : 1256871269331662
depositFeeBP *uint16* : 0

Pool with id 5:

lpToken *address* : 0x94026f0227cE0c9611e8a228f114F9F19CC3Fa87
allocPoint *uint256* : 200
lastRewardBlock *uint256* : 6558117
accNeonPerShare *uint256* : 2291138529696
depositFeeBP *uint16* : 0

Pool with id 6:

lpToken *address* : 0x1B96B92314C44b159149f7E0303511fB2Fc4774f
allocPoint *uint256* : 1
lastRewardBlock *uint256* : 6558040
accNeonPerShare *uint256* : 46183907807304981817891944858 56
depositFeeBP *uint16* : 5000

Pool with id 7:

lpToken *address* : 0x0Ed8E0A2D99643e1e65CCA22Ed4424090B8B7458
allocPoint *uint256* : 1
lastRewardBlock *uint256* : 6557825
accNeonPerShare *uint256* : 46167996268491715533661696813 58
depositFeeBP *uint16* : 5000

Pool with id 8:

lpToken *address* : 0xd9A0d1F5e02dE2403f68Bb71a15F8847A854b494
allocPoint *uint256* : 1
lastRewardBlock *uint256* : 6558039
accNeonPerShare *uint256* : 46128945971721905227010418966 74
depositFeeBP *uint16* : 5000

Pool with id 9:

lpToken *address* : 0xb8875e207EE8096a929D543C9981C9586992eAcb
allocPoint *uint256* : 1
lastRewardBlock *uint256* : 6557827
accNeonPerShare *uint256* : 46050845378172392707514587260 30
depositFeeBP *uint16* : 5000

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.	Passed
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.	Passed
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

No medium severity issues found.

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.

```
function updateEmissionRate(uint256 _neonPerBlock) public onlyOwner {
    require(
        _neonPerBlock <= NEON_PER_BLOCK_MAX,
        "updateEmissionRate: invalid _neonPerBlock value"
    );
    massUpdatePools();
    neonPerBlock = _neonPerBlock;
}
```

Owner privileges

1. Owner privileges

- Owner can change the pool details using a function set.

```
function set(
    uint256 _pid,
    uint256 _allocPoint,
    uint16 _depositFeeBP
) public onlyOwner {
    require(
        _depositFeeBP <= 10000,
        "set: invalid deposit fee basis points"
    );
    massUpdatePools();
    totalAllocPoint = totalAllocPoint.sub(poolInfo[_pid].allocPoint).add(
        _allocPoint
    );
    poolInfo[_pid].allocPoint = _allocPoint;
    poolInfo[_pid].depositFeeBP = _depositFeeBP;
}
```

- Owner can change the holder's address.

```
function holders(address _holdersAddress) public onlyOwner {
    holdersAddress = _holdersAddress;
}
```

- Owner can turn on / turn off the automatic emission feature.

```
function setUpdateDecreaseEmissionRateAutomatically(
    bool _decreaseEmissionRateAutomatically
) public onlyOwner {
    lastUpdateEmissionRate = block.timestamp;
    decreaseEmissionRateAutomatically = _decreaseEmissionRateAutomatically;
}
```

Notes

- There are no delegates moving to the zero address after the burn in Token contract.
- One percent from each transfer will be burnt.

Conclusion

Smart contracts do not contain any high severity issues! However, there are some owner privileges.

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.