



TechRate
AUDIT COMPANY

Smart Contract Security Audit

Audit Details



Audited project

Netefi Network Token



Deployer address

0x85641f72a23145834faa600dd1b64ae45edec1db



Client contacts:

Netefi Network Token team



Blockchain

Binance Smart Chain



Project website:

<https://www.netefi.com>

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 Netefi Network Token to perform an audit of smart contracts:

<https://bscscan.com/address/0xc77048e74a9c3f65ccc8b2d05f97104106a89b29#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 27.09.2021

Contract name	Netefi Network Token
Contract address	0xc77048e74a9C3F65cCC8B2d05F97104106A89b29
Total supply	15,725,240.6384
Token ticker	NEFI
Decimals	18
Token holders	283
Transactions count	1,435
Top 100 holders dominance	97.12%
Pause	False
Gov address	0x00
Locker address	0x00
Partner address	0x333dc90b373c8e78e42535093731965c323e9cc1
Contract deployer address	0x85641f72a23145834faa600dd1b64ae45edec1db
Contract's current owner address	0x85641f72a23145834faa600dd1b64ae45edec1db

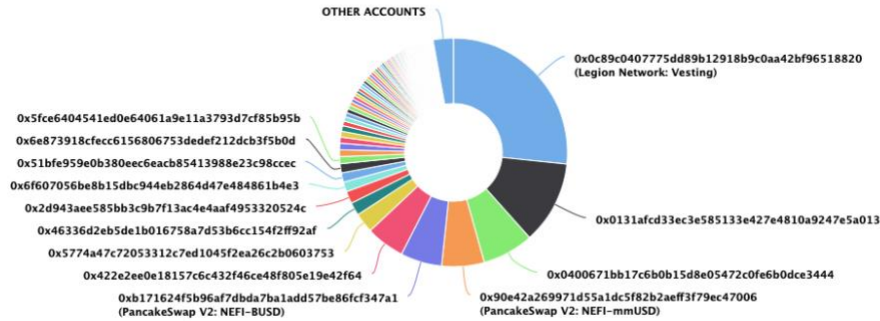
Netefi Network Token Token Distribution

The top 100 holders collectively own 97.12% (15,272,760.41 Tokens) of Netefi Network Token

Token Total Supply: 15,725,240.64 Token | Total Token Holders: 283

Netefi Network Token Top 100 Token Holders

Source: BscScan.com



(A total of 15,272,760.41 tokens held by the top 100 accounts from the total supply of 15,725,240.64 token)

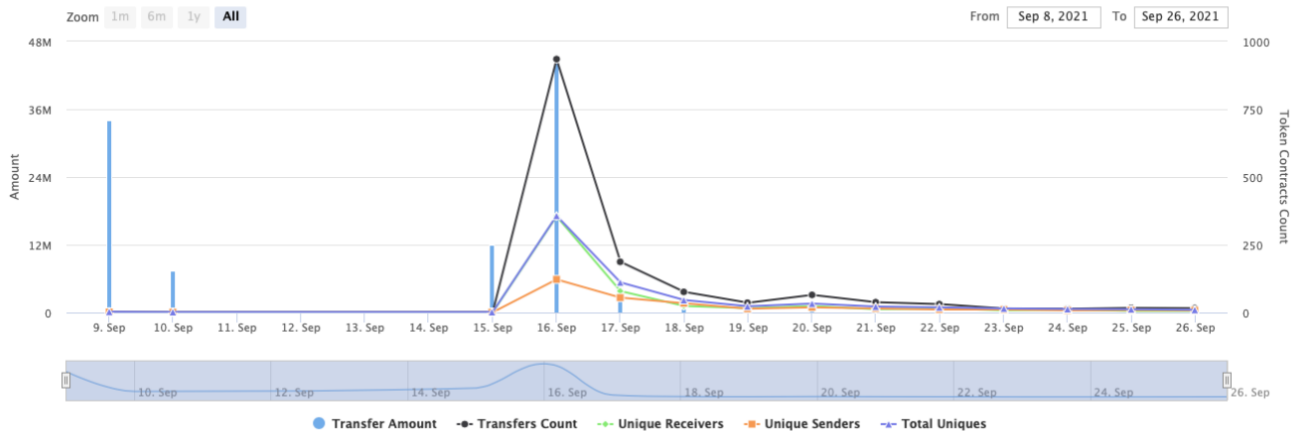
Netefi Network Token Contract Interaction Details

Time Series: Token Contract Overview





Thu 9, Sept 2021 - Sun 26, Sept 2021

Token Contract 0xc77048e74a9c3f65ccc8b2d05f97104106a89b29 (Netefi Network Token)

Source: BscScan.com



Netefi Network Token Top 10 Token Holders

Rank	Address	Quantity (Token)	Percentage
1	 Legion Network: Vesting	4,187,893.963333667	26.6317%
2	 0x0131afcd33ec3e585133e427e4810a9247e5a013	1,849,265.755095492853831644	11.7599%
3	0x0400671bb17c6b0b15d8e05472c0fe6b0dce3444	1,139,931.968860098429339592	7.2491%
4	 PancakeSwap V2: NEFI-mmUSD	948,115.282771987111982081	6.0293%
5	 PancakeSwap V2: NEFI-BUSD	924,032.164286167246351449	5.8761%
6	0x422e2ee0e18157c6c432f46ce48f805e19e42f64	860,001.968860098429339592	5.4689%
7	0x5774a47c72053312c7ed1045f2ea26c2b0603753	444,037.505594143618202691	2.8237%
8	0x46336d2eb5de1b016758a7d53b6cc154f2ff92af	289,846.701948827693966745	1.8432%
9	0x2d943aee585bb3c9b7f13ac4e4aaf4953320524c	272,580.532393826736423682	1.7334%
10	0x6f607056be8b15dbc944eb2864d47e484861b4e3	218,877.459310037308145785	1.3919%



Contract functions details

+ TimelockController (AccessControl)

- [Pub] <Constructor> #
- [Ext] <Fallback> (\$)
- [Pub] isOperation
- [Pub] isOperationPending
- [Pub] isOperationReady
- [Pub] isOperationDone
- [Pub] getTimestamp
- [Pub] getMinDelay
- [Pub] hashOperation
- [Pub] hashOperationBatch
- [Pub] schedule #
 - modifiers: onlyRole
- [Pub] scheduleBatch #
 - modifiers: onlyRole
- [Prv] _schedule #
- [Pub] cancel #
 - modifiers: onlyRole
- [Pub] execute (\$)
 - modifiers: onlyRoleOrOpenRole
- [Pub] executeBatch (\$)
 - modifiers: onlyRoleOrOpenRole
- [Prv] _beforeCall
- [Prv] _afterCall #
- [Prv] _call #
- [Ext] updateDelay #

+ [Lib] Strings

- [Int] toString
- [Int] toHexString
- [Int] toHexString

+ [Lib] SafeCast

- [Int] toUint224
- [Int] toUint128
- [Int] toUint96
- [Int] toUint64
- [Int] toUint32
- [Int] toUint16
- [Int] toUint8
- [Int] toUint256
- [Int] toInt128
- [Int] toInt64
- [Int] toInt32
- [Int] toInt16
- [Int] toInt8
- [Int] toInt256

+ Pausable (Context)

- [Pub] <Constructor> #
- [Pub] paused

- [Int] _pause #
 - modifiers: whenNotPaused
- [Int] _unpause #
 - modifiers: whenPaused
- + Ownable (Context)
 - [Pub] <Constructor> #
 - [Pub] owner
 - [Pub] renounceOwnership #
 - modifiers: onlyOwner
 - [Pub] transferOwnership #
 - modifiers: onlyOwner
 - [Prv] _setOwner #
- + [Lib] Math
 - [Int] max
 - [Int] min
 - [Int] average
 - [Int] ceilDiv
- + [Int] IERC20Metadata (IERC20)
 - [Ext] name
 - [Ext] symbol
 - [Ext] decimals
- + [Int] IERC20
 - [Ext] totalSupply
 - [Ext] balanceOf
 - [Ext] transfer #
 - [Ext] allowance
 - [Ext] approve #
 - [Ext] transferFrom #
- + [Int] IERC165
 - [Ext] supportsInterface
- + [Int] IAccessControl
 - [Ext] hasRole
 - [Ext] getRoleAdmin
 - [Ext] grantRole #
 - [Ext] revokeRole #
 - [Ext] renounceRole #
- + ERC20Votes (ERC20Permit)
 - [Pub] checkpoints
 - [Pub] numCheckpoints
 - [Pub] delegates
 - [Pub] getVotes
 - [Pub] getPastVotes
 - [Pub] getPastTotalSupply
 - [Prv] _checkpointsLookup
 - [Pub] delegate #
 - [Pub] delegateBySig #
 - [Int] _maxSupply
 - [Int] _mint #

- [Int] _burn #
- [Int] _afterTokenTransfer #
- [Int] _delegate #
- [Prv] _moveVotingPower #
- [Prv] _writeCheckpoint #
- [Prv] _add
- [Prv] _subtract

- + ERC20Burnable (Context, ERC20)
 - [Pub] burn #
 - [Pub] burnFrom #

- + ERC20 (Context, IERC20, IERC20Metadata)
 - [Pub] <Constructor> #
 - [Pub] name
 - [Pub] symbol
 - [Pub] decimals
 - [Pub] totalSupply
 - [Pub] balanceOf
 - [Pub] transfer #
 - [Pub] allowance
 - [Pub] approve #
 - [Pub] transferFrom #
 - [Pub] increaseAllowance #
 - [Pub] decreaseAllowance #
 - [Int] _transfer #
 - [Int] _mint #
 - [Int] _burn #
 - [Int] _approve #
 - [Int] _beforeTokenTransfer #
 - [Int] _afterTokenTransfer #

- + ERC165 (IERC165)
 - [Pub] supportsInterface

- + [Lib] ECDSA
 - [Prv] _throwError
 - [Int] tryRecover
 - [Int] recover
 - [Int] tryRecover
 - [Int] recover
 - [Int] tryRecover
 - [Int] recover
 - [Int] toEthSignedMessageHash
 - [Int] toTypedDataHash

- + [Int] IERC20Permit
 - [Ext] permit #
 - [Ext] nonces
 - [Ext] DOMAIN_SEPARATOR

- + ERC20Permit (ERC20, IERC20Permit, EIP712)
 - [Pub] <Constructor> #
 - modifiers: EIP712
 - [Pub] permit #

- [Pub] nonces
- [Ext] DOMAIN_SEPARATOR
- [Int] _useNonce #
- + EIP712
 - [Pub] <Constructor> #
 - [Int] _domainSeparatorV4
 - [Prv] _buildDomainSeparator
 - [Int] _hashTypedDataV4
- + [Lib] Counters
 - [Int] current
 - [Int] increment #
 - [Int] decrement #
 - [Int] reset #
- + Context
 - [Int] _msgSender
 - [Int] _msgData
- + AccessControl (Context, IAccessControl, ERC165)
 - [Pub] supportsInterface
 - [Pub] hasRole
 - [Int] _checkRole
 - [Pub] getRoleAdmin
 - [Pub] grantRole #
 - modifiers: onlyRole
 - [Pub] revokeRole #
 - modifiers: onlyRole
 - [Pub] renounceRole #
 - [Int] _setupRole #
 - [Int] _setRoleAdmin #
 - [Prv] _grantRole #
 - [Prv] _revokeRole #
- + NEFI (ERC20Votes, ERC20Burnable, Pausable, Ownable)
 - [Pub] <Constructor> #
 - modifiers: ERC20,ERC20Permit
 - [Pub] mint #
 - modifiers: whenNotPaused,onlyUnlocked,onlyGov
 - [Pub] mintFromLocker #
 - modifiers: whenNotPaused,whenPassMinLock,onlyNetefiLocker
 - [Pub] setupPartnerAddress #
 - modifiers: onlyOwner
 - [Pub] setupLockerAddress #
 - modifiers: onlyOwner
 - [Pub] setupGovAddress #
 - modifiers: onlyOwner
 - [Pub] unlockFromPartner #
 - modifiers: onlyPartner
 - [Pub] unlockFromOwner #
 - modifiers: onlyOwner
 - [Pub] burn #
 - [Pub] burnFrom #
 - [Int] _afterTokenTransfer #

- [Int] _mint #
- [Int] _burn #
- [Pub] pause #
 - modifiers: onlyOwner
- [Pub] unpause #
 - 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.	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

No low severity issues found.

Owner privileges (In the period when the owner is not renounced)

- Owner can change partner, locker and gov addresses.

```
fttrace | funcSig
function setupPartnerAddress(address _partnerAddress↑) public onlyOwner {
    require(
        partnerAddress == address(0) && _partnerAddress↑ != address(0),
        "NEFI: partner has set"
    );
    partnerAddress = _partnerAddress↑;
}

fttrace | funcSig
function setupLockerAddress(address _lockerAddress↑) public onlyOwner {
    require(
        lockerAddress == address(0) && _lockerAddress↑ != address(0),
        "NEFI: locker has set"
    );
    lockerAddress = _lockerAddress↑;
}

fttrace | funcSig
function setupGovAddress(address _govAddress↑) public onlyOwner {
    require(
        govAddress == address(0) && _govAddress↑ != address(0),
        "NEFI: governor has set"
    );
    govAddress = _govAddress↑;
}
```

- Owner can enable isOwnerUnlocked variable.

```
function unlockFromOwner() public onlyOwner {
    require(isOwnerUnlocked == false, "NEFI: owner done unlocked");
    isOwnerUnlocked = true;
}
```


- Owner can pause/unpause contract.

```
ftrace | funcSig
function pause() public onlyOwner {
    _pause();
}

ftrace | funcSig
function unpause() public onlyOwner {
    _unpause();
}
```

- Owner can enable isPartnerUnlocked variable.

```
function unlockFromPartner() public onlyPartner {
    require(isPartnerUnlocked == false, "NEFI: partner done unlocked");
    isPartnerUnlocked = true;
}
```

- lockerAddress can mint and reduce LOCKED_AMOUNT.

```
function mintFromLocker(address to↑, uint256 amount↑)
    public
    whenNotPaused
    whenPassMinLock
    onlyNetefiLocker
{
    require(to↑ != address(0), "NEFI: to address invalid");
    require(amount↑ > 0 && amount↑ <= LOCKED_AMOUNT, "NEFI: amount invalid");
    super._mint(to↑, amount↑);
    LOCKED_AMOUNT = LOCKED_AMOUNT - amount↑;
}
```

- Gov address can mint under (_maxSupply() - LOCKED_AMOUNT).

```
function mint(address to↑, uint256 amount↑)
    public
    whenNotPaused
    onlyUnlocked
    onlyGov
{
    require(to↑ != address(0), "NEFI: to address invalid");
    require(amount↑ > 0, "NEFI: amount invalid");
    require(
        (totalSupply() + amount↑) <= (_maxSupply() - LOCKED_AMOUNT),
        "NEFI: exceeded mint quota"
    );
    super._mint(to↑, amount↑);
}
```

- Owner can enable isPartnerUnlocked variable.

```
function unlockFromPartner() public onlyPartner {
    require(isPartnerUnlocked == false, "NEFI: partner done unlocked");
    isPartnerUnlocked = true;
}
```

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.