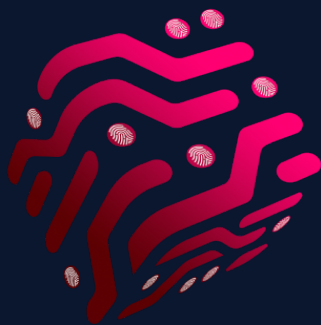




# SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS REPORT



**3NT**  
\$3NT

**29/04/2024**



# TOKEN OVERVIEW

---

## Fees

- Buy fees: 2%
- Sell fees: 1%

## Fees privileges

- Can change buy fees up to 10% and sell fees up to 10%

## Ownership

- Owned

## Minting

- No mint function

## Max Tx Amount / Max Wallet Amount

- Can change max tx amount and max wallet amount (with threshold)

## Blacklist

- Blacklist function not detected

## Other privileges

- Can exclude / include from fees, wallet limitations and tx limitations
  - Investors won't be able to sell their tokens on PancakeSwap if swapEnabled is true and thresholdSwapAmount is set to 0
  - Contract owner has to set swapTrading function to it's true value
-

# TABLE OF CONTENTS

1

DISCLAIMER

2

INTRODUCTION

3

WEBSITE + SOCIALS

4-5

AUDIT OVERVIEW

6-9

OWNER PRIVILEGES

10

CONCLUSION AND ANALYSIS

11

TOKEN DETAILS

12

3NT ANALYTICS &  
TOP 10 TOKEN HOLDERS

13

TECHNICAL DISCLAIMER



# DISCLAIMER

The information provided on this analysis document is only for general information and should not be used as a reason to invest.

FreshCoins Team will take no payment for manipulating the results of this audit.

The score and the result will stay on this project page information on our website <https://freshcoins.io>

FreshCoins Team does not guarantees that a project will not sell off team supply, or any other scam strategy ( RUG or Honeypot etc )



# INTRODUCTION

**FreshCoins** (Consultant) was contracted by **3NT** (Customer) to conduct a Smart Contract Code Review and Security Analysis.

**0xd6262fad97c1d7832a1bceb47d93fae1423327e7**

Network: **Base Network (ETH)**

This report presents the findings of the security assessment of Customer's smart contract and its code review conducted on **29/04/2024**



# WEBSITE DIAGNOSTIC

<https://nnn.cloud>



0-49



50-89



90-100



Performance



Accessibility



Best  
Practices



SEO



Progressive  
Web App

## Socials



Twitter

N/A



Telegram

N/A

# AUDIT OVERVIEW



Security Score  
**HIGH RISK**  
Audit FAIL



**Static Scan**  
Automatic scanning for  
common vulnerabilities



**ERC Scan**  
Automatic checks for  
ERC's conformance



High



Medium



Low



Optimizations



Informational



No.	Issue description	Checking Status
1	Compiler Errors / Warnings	Passed
2	Reentrancy and Cross-function	Passed
3	Front running	Low
4	Timestamp dependence	Passed
5	Integer Overflow and Underflow	Passed
6	Reverted DoS	Passed
7	DoS with block gas limit	Passed
8	Methods execution permissions	Passed
9	Exchange rate impact	Passed
10	Malicious Event	Passed
11	Scoping and Declarations	Passed
12	Uninitialized storage pointers	Passed
13	Design Logic	Passed
14	Safe Zeppelin module	Passed



# OWNER PRIVILEGES

- Contract owner can't mint tokens after initial contract deploy

- Contract owner can't exclude addresses from transactions

- Contract owner can change swap settings (without threshold)

Users won't be able to sell their tokens on PancakeSwap if swapEnabled is true and thresholdSwapAmount is set to 0 because every transaction involving the token will trigger a swap for ETH, regardless of the transaction size

Default value: uint256 private thresholdSwapAmount = totalSupply \* 4 / 10000;

```
function updateThresholdSwapAmount(uint256 newAmount) external onlyOwner returns(bool){
    thresholdSwapAmount = newAmount;
    return true;
}
```

- Contract owner has to set swapTrading function to it's true value

Please note that any wallet excluded from tax retains the ability to engage in trading, even in situations where trading has been disabled

```
function swapTrading() external onlyOwner {
    isTrading = true;
    swapEnabled = true;
    taxTill = block.number + 10;
}

_transferFrom function line 700
.
.
.
if (!isTrading) {
    require(!_isExcludedFromFees[sender] || !_isExcludedFromFees[recipient], "Trading is not active.");
}
.
.
.
```

- Contract owner can change max tx amount limitation (with threshold)

```
function updateMaxTxnAmount(uint256 newMaxBuy, uint256 newMaxSell) external onlyOwner {
    require(((totalSupply() * newMaxBuy) / 1000) >= (totalSupply() / 100), "maxBuyAmount must be higher than 1%");
    require(((totalSupply() * newMaxSell) / 1000) >= (totalSupply() / 100), "maxSellAmount must be higher than 1%");
    maxBuyAmount = (totalSupply() * newMaxBuy) / 1000;
    maxSellAmount = (totalSupply() * newMaxSell) / 1000;
}
```

## ● Contract owner can change max wallet amount limitation (with threshold)

maxWalletAmount should be at least 2% of totalSupply

```
function updateMaxWalletAmount(uint256 newPercentage) external onlyOwner {
    require(((totalSupply() * newPercentage) / 1000) >= (totalSupply() / 100), "Cannot set maxWallet lower than 1%");
    maxWalletAmount = (totalSupply() * newPercentage) / 1000;
}
```

## ● Contract owner can change buy fees up to 10% and sell fees up to 10%

```
function updateFees(uint8 _marketingFeeBuy, uint8 _liquidityFeeBuy, uint8 _devFeeBuy, uint8 _marketingFeeSell, uint8 _liquidityFeeSell, uint8 _devFeeSell) external onlyOwner {
    _fees.buyMarketingFee = _marketingFeeBuy;
    _fees.buyLiquidityFee = _liquidityFeeBuy;
    _fees.buyDevFee = _devFeeBuy;
    _fees.buyTotalFees = _fees.buyMarketingFee + _fees.buyLiquidityFee + _fees.buyDevFee;

    _fees.sellMarketingFee = _marketingFeeSell;
    _fees.sellLiquidityFee = _liquidityFeeSell;
    _fees.sellDevFee = _devFeeSell;
    _fees.sellTotalFees = _fees.sellMarketingFee + _fees.sellLiquidityFee + _fees.sellDevFee;
    require(_fees.buyTotalFees <= 10, "Must keep fees at 30% or less");
    require(_fees.sellTotalFees <= 10, "Must keep fees at 30% or less");
}
```

## ● Contract owner can exclude/include wallet from tax

```
function excludeFromFees(address account, bool excluded) public onlyOwner {
    _isExcludedFromFees[account] = excluded;
}
```

## ● Contract owner can exclude/include wallet from wallet limitations

```
function excludeFromWalletLimit(address account, bool excluded) public onlyOwner {
    _isExcludedMaxWalletAmount[account] = excluded;
}
```

## ● Contract owner can exclude/include wallet from tx limitations

```
function excludeFromMaxTransaction(address updAds, bool isEx) public onlyOwner {
    _isExcludedMaxTransactionAmount[updAds] = isEx;
}
```

## ● Contract owner has ability to retrieve any token held by the contract

Native tokens NOT excluded

```
function rescueETH(uint256 weiAmount) external onlyOwner {
    payable(owner()).transfer(weiAmount);
}

function rescueERC20(address tokenAdd, uint256 amount) external onlyOwner {
    IERC20(tokenAdd).transfer(owner(), amount);
}
```

## ● Contract owner can change marketingWallet and devWallet addresses

Default values:

marketingWallet: 0xb5faf9441eFF44801F3450214ddE135836003DAA

devWallet: 0xe9A37A1524ec504c940Fb9805bC41B58fbE23819

Ensure that \_devWallet and \_marketingWallet is not a dead address or a contract address

```
function setWallets(address _marketingWallet,address _devWallet) external onlyOwner{
    marketingWallet = _marketingWallet;
    devWallet = _devWallet;
}
```

## ● Contract owner can transfer ownership

```
function transferOwnership(address newOwner) public virtual onlyOwner {
    require(newOwner != address(0), "Ownable: new owner is the zero address");
    emit OwnershipTransferred(_owner, newOwner);
    _owner = newOwner;
}
```

## ● Contract owner can renounce ownership

```
function renounceOwnership() public virtual onlyOwner {
    emit OwnershipTransferred(_owner, address(0));
    _owner = address(0);
}
```

### Recommendation:

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. The risk can be prevented by temporarily locking the contract or renouncing ownership.



# CONCLUSION AND ANALYSIS



Smart Contracts within the scope were manually reviewed and analyzed with static tools.



Audit report overview contains all found security vulnerabilities and other issues in the reviewed code.



Found 2 HIGH issues during the first review.

# TOKEN DETAILS

## Details

Buy fees:	2%
Sell fees:	1%
Max TX:	40,000,000
Max Wallet:	40,000,000

## Honeypot Risk

Ownership:	Owned
Blacklist:	Not detected
Modify Max TX:	Detected
Modify Max Sell:	Detected
Disable Trading:	Not detected

## Rug Pull Risk

Liquidity:	N/A
Holders:	100% unlocked tokens



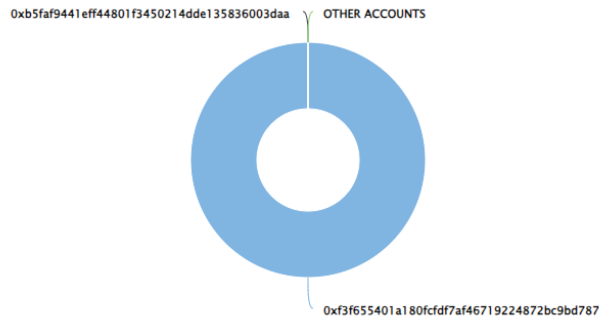
# 3NT TOKEN ANALYTICS & TOP 10 TOKEN HOLDERS

The top 10 holders collectively own 100.00% (1,000,000,000.00 Tokens) of 3NT

Token Total Supply: 1,000,000,000.00 Token | Total Token Holders: 2

## 3NT Top 10 Token Holders

Source: <https://basescan.org>



(A total of 1,000,000,000.00 tokens held by the top 10 accounts from the total supply of 1,000,000,000.00 token)

Rank	Address	Quantity (Token)	Percentage
1	0xf3f655401a180fcdf7af46719224872bc9bd787	999,999,900	100.0000%
2	0xb5faf9441eff44801f3450214dde135836003daa	100	0.0000%

# TECHNICAL DISCLAIMER

Smart contracts are deployed and executed on the blockchain platform. The platform, its programming language, and other software related to the smart contract can have its vulnerabilities that can lead to hacks. The audit can't guarantee the explicit security of the audited project / smart contract.

