



## SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS REPORT



**W3INU (W3INU.COM)**  
**\$W3INU**



**12/02/2022**



# TABLE OF CONTENTS

- 1 DISCLAIMER
- 2 INTRODUCTION
- 3-4 WEBSITE DIAGNOSTIC
- 5-6 AUDIT OVERVIEW
- 7-9 OWNER PRIVILEGES
- 10 CONCLUSION AND ANALYSIS
- 11 TOKEN DETAILS
- 12 W3INU (W3INU.COM) TOKEN 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 W3INU (W3INU.COM) (Customer) to conduct a Smart Contract Code Review and Security Analysis.

0x5609ab73D9119Cc5611c10c90958CB792f8482Cf

Network: Binance Smart Chain (BNB)

This report presents the findings of the security assessment of Customer's smart contract and its code review conducted on 12/02/2022



# WEBSITE DIAGNOSTIC

<https://w3inu.com/>



0-49



50-89



90-100



Performance



Accessibility



Best Practices



SEO



Progressive  
Web App

## Metrics



First Contentful Paint

**2.8 s**



Time to interactive

**5.1 s**



Speed Index

**12.1 s**



Total Blocking Time

**24 ms**



Large Contentful Paint

**10.6 s**



Cumulative Layout Shift

**0**

# WEBSITE IMPROVEMENTS

---

Reduce initial server response time

---

Reduce unused CSS

---

Reduce unused JavaScript

---

Image elements do not have explicit `width` and `height`

---

Ensure text remains visible during webfont load

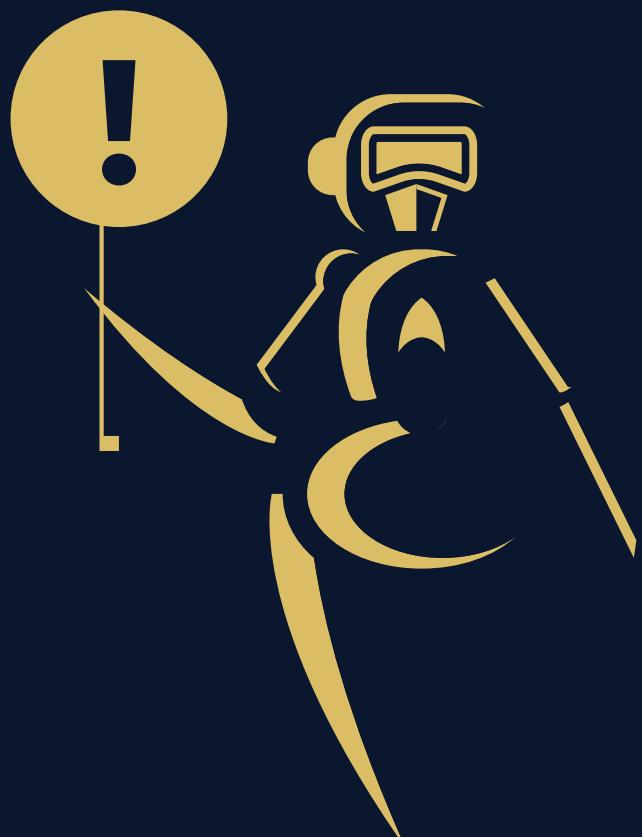
---

Background and foreground colors do not have a sufficient contrast ratio

---

Heading elements are not in a sequentially-descending order

---



# AUDIT OVERVIEW



**Security Score**



**Static Scan**  
Automatic scanning for common vulnerabilities



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

0 **High**

0 **Medium**

0 **Low**

0 **Optimizations**

0 **Informational**



No.	Issue description	Checking Status
1	Compiler Errors / Warnings	Passed
2	Reentrancy and Cross-function	Passed
3	Front running	Passed
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 exclude an address from transactions.

Contract owner can't mint tokens after initial contract deploy

Contract owner can exclude/include wallet from fees

```
function setExcludeFromFee(address account, bool value) external onlyOwner {
    require(_isExcludedFromFee[account] != value, "Already set");
    _isExcludedFromFee[account] = value;
}
```

Contract owner can exclude/include wallet from reward

```
function excludeFromReward(address account) public onlyOwner {
    require(!_isExcludedFromReward[account], "Account is already excluded");
    if (_reflectionBalance[account] > 0) {
        _tokenBalance[account] = tokenFromReflection(
            _reflectionBalance[account]
        );
    }
    _isExcludedFromReward[account] = true;
    _excludedFromReward.push(account);
}

function includeInReward(address account) external onlyOwner {
    require(_isExcludedFromReward[account], "Account is already included");
    for (uint256 i = 0; i < _excludedFromReward.length; i++) {
        if (_excludedFromReward[i] == account) {
            _excludedFromReward[i] = _excludedFromReward[_excludedFromReward.length - 1];
            _tokenBalance[account] = 0;
            _isExcludedFromReward[account] = false;
            _excludedFromReward.pop();
            break;
        }
    }
}
```

Contract owner can change max tx amount

```
function setMaxTxAmount(uint256 _newAmount) external onlyOwner() {
    require(_newAmount >= 1 * 10**18, "maxTxAmount should be greater than 1 token");
    maxTxAmount = _newAmount;
}
```

## Contract owner can change `farmAndBuybackWallet`, `marketingWallet`, `devWallet` addresses

Current values:

`farmAndBuybackWallet`: `0xf2c38e8461c88f5bcd9064978ffaedee9cf30b37`  
`marketingWallet`: `0xa27645c48d6e8e1ac9380d8deaa24a2c75b0ef44`  
`devWallet`: `0x7313d62c0c4b09746ce6301fb7ef4c94ae619768`

```
function setFarmAndBuybackWallet(address _address) external onlyOwner {
    farmAndBuybackWallet = _address;
}

function setMarketingWallet(address _address) external onlyOwner {
    marketingWallet = _address;
}

function setDevWallet(address _address) external onlyOwner {
    devWallet = _address;
}
```

## Contract owner can enable/disable fees

```
function setFeeActive(bool value) external onlyOwner{
    require(isFeeActive != value, "Already set");
    isFeeActive = value;
}
```

## Contract owner can change swap settings

```
function setSwapEnabled(bool enabled) external onlyOwner {
    require(swapEnabled != enabled, "Already set");
    swapEnabled = enabled;
}

.

.

.

function setMinTokensBeforeSwap(uint256 _amount) external onlyOwner {
    minTokensBeforeSwap = _amount;
}
```

## 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);  
}
```

## Contract owner can change the fees

```
function setTransferFee(uint256 buy, uint256 sell, uint256 p2p) external onlyOwner {  
    require(buy <= 2500 && sell <= 2500 && p2p <= 2500, "Invalid fee");  
    _transferFee[0] = buy;  
    _transferFee[1] = sell;  
    _transferFee[2] = p2p;  
}  
  
function setMarketingFee(uint256 buy, uint256 sell, uint256 p2p) external onlyOwner {  
    require(buy <= 2500 && sell <= 2500 && p2p <= 2500, "Invalid fee");  
    _marketingFee[0] = buy;  
    _marketingFee[1] = sell;  
    _marketingFee[2] = p2p;  
}  
  
function setFarmAndBuybackFee(uint256 buy, uint256 sell, uint256 p2p) external onlyOwner {  
    require(buy <= 2500 && sell <= 2500 && p2p <= 2500, "Invalid fee");  
    _farmAndBuybackFee[0] = buy;  
    _farmAndBuybackFee[1] = sell;  
    _farmAndBuybackFee[2] = p2p;  
}
```

### 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 no issue during the first review.

# TOKEN DETAILS

## Details

Buy fees: 10%

Sell fees: 10%

Max TX: N/A

Max Sell: N/A

## Honeypot Risk

Ownership: Owned

Blacklist: Not detected

Modify Max TX: Detected

Modify Max Sell: Not detected

Disable Trading: Not detected

## Rug Pull Risk

Liquidity: N/A

Holders: Clean



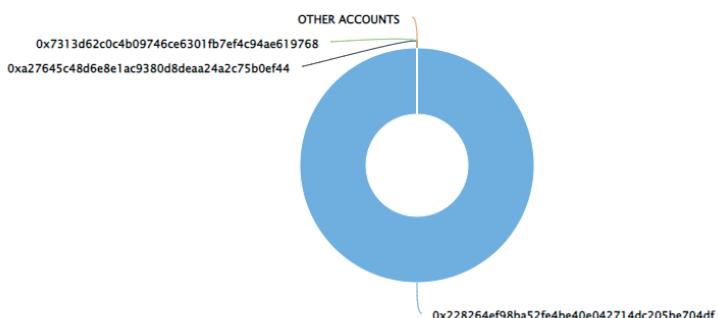
# W3INU (W3INU.COM) TOKEN ANALYTICS & TOP 10 TOKEN HOLDERS

The top 10 holders collectively own 100.00% (1,000,000,000,000.00 Tokens) of W3INU (W3INU.COM)

Token Total Supply: 1,000,000,000,000.00 Token | Total Token Holders: 3

W3INU (W3INU.COM) Top 10 Token Holders

Source: BscScan.com



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

Rank	Address	Quantity (Token)	Percentage
1	0x228264ef98ba52fe4be40e042714dc205be704df	999,998,000,000	99.9998%
2	0xa27645c48d6e8e1ac9380d8deaa24a2c75b0ef44	1,000,000	0.0001%
3	0x7313d62c0c4b09746ce6301fb7ef4c94ae619768	1,000,000	0.0001%

# 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.

