



# SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS REPORT



WALL STREET DECENTRAL (WALLCOIN.IO)  
\$WALL



17/02/2022



# 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 WALL STREET DECENTRAL TOKEN DISTRIBUTION & 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  
**WALL STREET DECENTRAL (WALLCOIN.IO) (Customer)** to conduct a Smart Contract Code Review and Security Analysis.

0x251A67e577DBbCFB9bF79E4602C3f1c11cc869e0

Network: **Binance Smart Chain (BSC)**

This report presents the findings of the security assessment of Customer's smart contract and its code review conducted on **17/02/2022**



# WEBSITE DIAGNOSTIC

<https://wallcoin.io/>



0-49



50-89



90-100



Performance



Accessibility



Best Practices



SEO



Progressive  
Web App

## Metrics



First Contentful Paint

**3.1 s**



Time to interactive

**10.2 s**



Speed Index

**4.2 s**



Total Blocking Time

**820 ms**



Large Contentful Paint

**12.4 s**



Cumulative Layout Shift

**0.002**

# WEBSITE IMPROVEMENTS

---

Reduce initial server response time

---

Reduce unused CSS

---

Reduce unused JavaScript

---

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

---

Reduce JavaScript execution time `1.6 s`

---

Image elements do not have `[alt]` attributes

---

Links do not have a discernible name

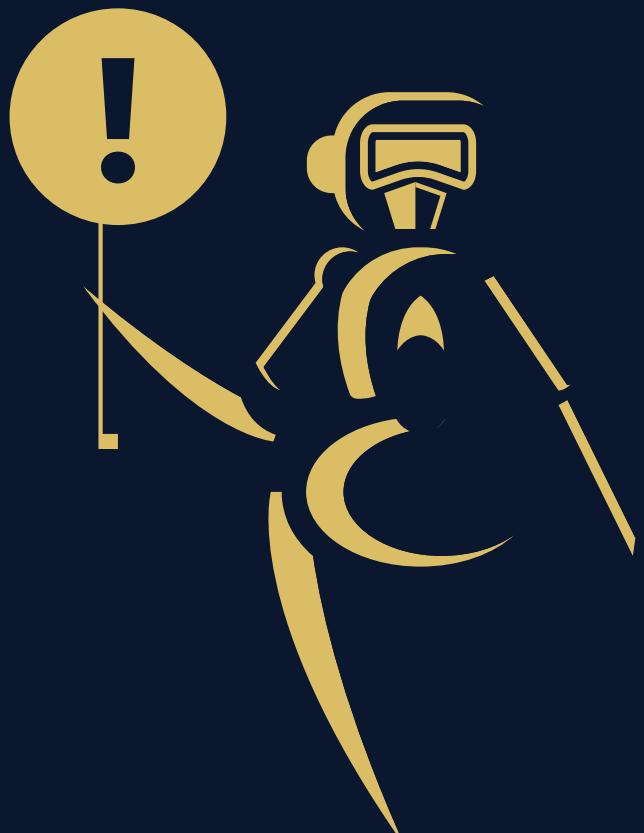
---

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



High



Medium



Low



Optimizations



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       | Low             |
| 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 exclude/include wallet from fees

```
function excludeFromFee(address account) public onlyOwner() {
    _isExcludedFromFee[account] = true;
}

function includeInFee(address account) public onlyOwner() {
    _isExcludedFromFee[account] = false;
}
```

Contract owner can exclude/include wallet from rewards

```
function excludeFromReward(address account) external onlyOwner() {
    require(!_isExcluded[account], "Account is already excluded");
    if(_rOwned[account] > 0) {
        _tOwned[account] = tokenFromReflection(_rOwned[account]);
    }
    _isExcluded[account] = true;
    _excluded.push(account);
}

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

Contract owner can change buyback settings

```
function setBuyback(bool b) external onlyOwner() {
    _enableBuyback = b;
}

function setBuybackBNBThreshold(uint256 bnbAmount) external onlyOwner() {
    _buybackBNBThreshold = bnbAmount;
}

function setBuybackUpperLimit(uint256 buybackLimit) external onlyOwner() {
    _buybackUpperLimit = buybackLimit;
}

function setBuybackBNBPercentage(uint256 percentage) external onlyOwner() {
    _buybackBNBPercentage = percentage;
}
```

## Contract owner can change swap settings

```
function setLiquidity(bool b) external onlyOwner() {  
    _enableLiquidity = b;  
}
```

## Contract owner can change lottery settings

```
function setLottery(bool b) external onlyOwner() {  
    _enableLottery = b;  
}  
  
function setLotteryChance(uint chance) external onlyOwner() {  
    _lotteryChance = chance;  
}  
  
function setLotteryThreshold(uint256 threshold) external onlyOwner() {  
    _lotteryThreshold = threshold;  
}  
  
function setLotteryMinimumSpend(uint256 minimumSpend) external onlyOwner() {  
    _lotteryMinimumSpend = minimumSpend;  
}
```

## Contract owner can change `_marketingAddress` address

### Current address:

`_marketingAddress`: [0xb3c7b8987db782da25d943b2edd8245fa2234166](#)

```
function setMarketingAddress(address marketingAddress) external onlyOwner() {  
    _marketingAddress = payable(marketingAddress);  
}
```

## Contract owner can change max tx amount

```
function setMaxTxAmount(uint256 maxTxAmount) external onlyOwner() {  
    require(maxTxAmount >= (_tTotal.mul(1).div(10000)).div(10**18), "amount must be greater than 0.01%  
of the total supply");  
    _maxTxAmount = maxTxAmount;  
}
```

## Contract owner can renounce ownership

```
function renounceOwnership() public virtual onlyOwner {  
    _setOwner(address(0));  
}
```

## Contract owner can transfer ownership

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

## Contract owner can change the fees up to 50%

```
function setTaxFeePercent(uint256 taxFee) external onlyOwner() {
    require(taxFee <= 28, "Total fee is over 28%");
    _taxFee = taxFee;
}

function setSellTaxFeePerecent(uint256 taxFee) external onlyOwner() {
    require(taxFee <= 28, "Total fee is over 28%");
    _sellTaxFee = taxFee;
}

function setWhaleSellTaxFeePerecent(uint256 taxFee) external onlyOwner() {
    require(taxFee <= 50, "Total fee is over 50%");
    _whaleSellTaxFee = taxFee;
}
```

## Contract owner can change `_whaleSellThreshold`, `_tokenSwapThreshold` and `_whaleSellTimer`

```
function setTokenSwapThreshold(uint256 tokenSwapThreshold) external onlyOwner() {
    _tokenSwapThreshold = tokenSwapThreshold;
}

function setWhaleSellThreshold(uint256 amount) external onlyOwner() {
    _whaleSellThreshold = amount;
}

function setWhaleSellTimer(uint time) external onlyOwner() {
    _whaleSellTimer = time;
}
```

## Contract owner can withdraw BNB or non-W3ECO tokens from the contract

```
function withdrawBNB(uint256 amount) public onlyOwner() {
    if(amount == 0) payable(owner()).transfer(address(this).balance);
    else payable(owner()).transfer(amount);
}

function withdrawForeignToken(address token) public onlyOwner() {
    require(address(this) != address(token), "Cannot withdraw native token");
    IERC20(address(token)).transfer(msg.sender, IERC20(token).balanceOf(address(this)));
}
```

# 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 1 LOW issue during the first review.

# TOKEN DETAILS

## Details

|            |        |
|------------|--------|
| Buy fees:  | 7%     |
| Sell fees: | 14-21% |
| Max TX:    | N/A    |
| Max Sell:  | N/A    |

## 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:   | Clean |



# WALL STREET DECENTRAL TOKEN DISTRIBUTION & TOP 10 TOKEN HOLDERS



| Rank | Address                                    | Quantity (Token)           | Percentage |
|------|--|----------------------------|------------|
| 1    | 0x73fa3b86b27da17608cf79d325945c06657c6cce | 975,297,729,888,749.959102 | 97.5298%   |
| 2    | 0xc9da62823587d118fafca06ef763a8df530fb4e8 | 24,702,270,111,250.040898  | 2.4702%    |

# 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.

