# freshcoins

# SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS REPORT

## LADY BUG CHAIN
### $LADY

**22/12/2022**

# TOKEN OVERVIEW

## Fees

• Buy fees:          10%

• Sell fees:          10%

## Fees privileges

• Can change fees up to 100%

## Ownership

• Owned

## Minting

• No mint function

## Max Tx Amount / Max Wallet Amount

• Can change max tx amount without threshold and max wallet amount with threshold

## Blacklist

• Blacklist function detected

## Other privileges

• Can exclude / include from fees

# TABLE OF CONTENTS

# 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
**LADY BUG CHAIN** (Customer) to conduct a Smart Contract Code Review and
Security Analysis.

0x64b0a8b9633697EAF6B2B2d41A4196257cc8E1dd

**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 22/12/2022

# AUDIT OVERVIEW

**62**

**Security Score**

**93** Static Scan
Automatic scanning for
common vulnerabilities

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

**4** High

**1** 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 mint tokens after initial contract deploy**

● **Contract owner can exclude an address from transactions**

```
function isBots(address _address, bool _value) public onlyOwner{
    isBot[_address] = _value;
}
```

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

```
function setIsFeeExempt(address holder, bool exempt) external onlyOwner {
    isFeeExempt[holder] = exempt;
}
```

● **Contract owner can change fees up to 100%**

```
function setFees(uint256 _liquidityFee, uint256 _marketingFee, uint256 _devFee, uint256 _feeDenominator)
external onlyOwner {
    liquidityFee = _liquidityFee;
    marketingFee = _marketingFee;
    devFee = _devFee;
    totalFee = _liquidityFee.add(_marketingFee).add(_devFee);
    feeDenominator = _feeDenominator;
}
```

● **Contract owner can set sell multiplier** (without threshold)

```
function setSellMultiplier(uint256 multiplier) external onlyOwner{
    _sellMultiplier = multiplier;
}
```

● **Contract owner can change marketingFeeReceiver and devFeeReceiver addresses**

Current values:

marketingFeeReceiver : 0x3a62054c5696dce7a7365251a5d9cc50ff1dc953

devFeeReceiver : 0xff500963dc3e57e18819691adfe51a718038428a

```
function setFeeReceiver(address _marketingFeeReceiver, address _devFeeReceiver) external onlyOwner {
    marketingFeeReceiver = _marketingFeeReceiver;
    devFeeReceiver = _devFeeReceiver;
}
```

## ● Contract owner can change swap settings

```solidity
function setSwapBackSettings(bool _enabled, uint256 _amount) external onlyOwner {
    swapEnabled = _enabled;
    swapThreshold = _amount;
}
```

## ● Contract owner can change max tx amount limitation to 0

**Transfers can be disabled if _maxTxAmount value is set to 0**

```solidity
function setTxLimit(uint256 amountBuy) external onlyOwner {
    _maxTxAmount = amountBuy;
}
```

## ● Contract owner can change max wallet percent with threshold

```solidity
function setMaxWallet(uint256 amount) external onlyOwner {
    require(amount >= _totalSupply / 1000 );
    _maxWalletSize = amount;
}
```

## ● Contract owner can change cooldown between trades status and interval value

```solidity
function cooldownEnabled(bool _status, uint8 _interval) public onlyOwner() {
    opCooldownEnabled = _status;
    cooldownTimerInterval = _interval;
}
```

## ● Contract owner can withdraw stuck tokens from smart contract

```solidity
function transferForeignToken(address _token) public {
    require(_token != address(this), "Can't let you take all native token");
    uint256 _contractBalance = IBEP20(_token).balanceOf(address(this));
    payable(devFeeReceiver).transfer(_contractBalance);
}
```

## ● Contract owner can renounce ownership

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

## ● Contract owner can transfer ownership

```solidity
function transferOwnership(address payable adr) public onlyOwner {
    owner = adr;
    _intAddr[adr] = true;
    emit OwnershipTransferred(adr);
}
```

**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 4 HIGH issues during the first review.

# TOKEN DETAILS

## Details

| | |
|---|---|
| **Buy fees:** | 10% |
| **Sell fees:** | 10% |
| **Max TX:** | 100,000,000 |
| **Max Sell:** | N/A |

## Honeypot Risk

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

## Others

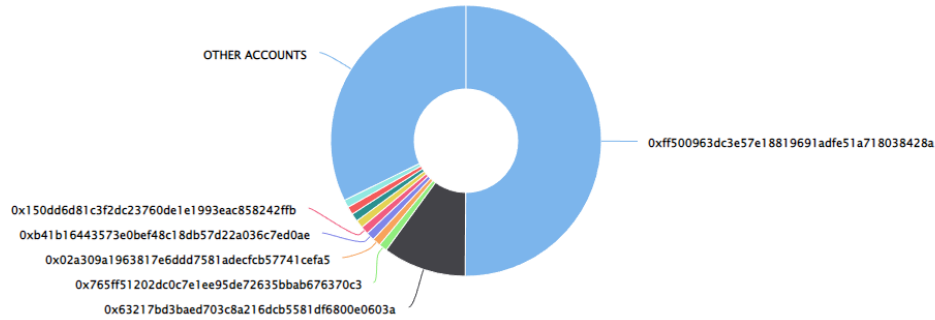| | |
|---|---|
| **Liquidity:** | N/A |
| **Holders:** | Clean |

# LADY TOKEN ANALYTICS
# & TOP 10 TOKEN HOLDERS

The top 10 holders collectively own 67.76% (6,775,924,619.41 Tokens) of LADY BUG CHAIN | Token Total Supply: 10,000,000,000.00 Token | Total Token Holders: 149

## LADY BUG CHAIN Top 10 Token Holders
Source: BscScan.com

OTHER ACCOUNTS

0xff500963dc3e57e18819691adfe51a718038428a

0x150dd6d81c3f2dc23760de1e1993eac858242ffb
0xb41b16443573e0bef48c18db57d22a036c7ed0ae
0x02a309a1963817e6ddd7581adecfcb57741cefa5
0x765ff51202dc0c7e1ee95de72635bbab676370c3
0x63217bd3baed703c8a216dcb5581df6800e0603a

(A total of 6,775,924,619.41 tokens held by the top 10 accounts from the total supply of 10,000,000,000.00 token)

| Rank | Address | Quantity (Token) | Percentage |
|------|---------|------------------|------------|
| 1 | 0xff500963dc3e57e18819691adfe51a718038428a | 5,009,324,664.952560162 | 50.0932% |
| 2 | 0x63217bd3baed703c8a216dcb5581df6800e0603a | 995,367,518.738687397 | 9.9537% |
| 3 | 0x765ff51202dc0c7e1ee95de72635bbab676370c3 | 99,000,000 | 0.9900% |
| 4 | 0x02a309a1963817e6ddd7581adecfcb57741cefa5 | 99,000,000 | 0.9900% |
| 5 | 0xb41b16443573e0bef48c18db57d22a036c7ed0ae | 99,000,000 | 0.9900% |
| 6 | 0x150dd6d81c3f2dc23760de1e1993eac858242ffb | 96,300,000 | 0.9630% |
| 7 | 0x7c09d5aa828d23b0a4189fa434777fa2c78e081a | 95,496,657.544705105 | 0.9550% |
| 8 | 0x4bbd9147575884532af67c7ad98988d0c66e5612 | 95,121,247.939882714 | 0.9512% |
| 9 | 0x70bab5fb9069da761547f6326db3cdc9b042b1c4 | 94,807,990.559565047 | 0.9481% |
| 10 | 0x8dca00e0b30518fca389669d21b7f6ba4f835f8e | 92,506,539.674290696 | 0.9251% |

# 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.