



freshcoins

SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS REPORT



Irish Shiba Coin
\$ISC



27/01/2022



TABLE OF CONTENTS

- 1 DISCLAIMER
- 2 INTRODUCTION
- 3-4 WEBSITE DIAGNOSTIC
- 5-6 AUDIT OVERVIEW
- 7-8 OWNER PRIVILEGES
- 9 CONCLUSION AND ANALYSIS
- 10 TOKEN DETAILS
- 11 IRISH SHIBA COIN TOKEN DISTRIBUTION & TOP 10 TOKEN HOLDERS
- 12 TECHNICAL DISCLAIMER



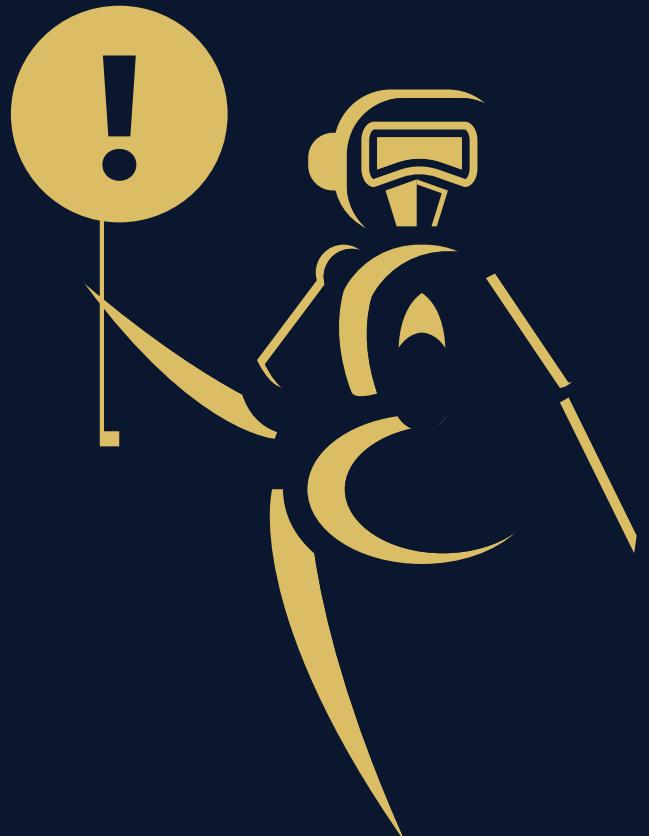
DISCLAIMER

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 Irish Shiba Coin (Customer) to conduct a Smart Contract Code Review and Security Analysis.

0x6e7F29c3717F7d03daC82Dd032EA43bE1ec45972

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 27/01/2022



AUDIT OVERVIEW



Security Score



Static Scan
Automatic scanning for common vulnerabilities



ERC Scan
Automatic checks for ERC's conformance

0 **High**

0 **Medium**

1 **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	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.

Owner must be a contract with transparent rules for using `setTaxes`

Contract owner can change the fees

```
function setFees(uint newLiquidityFee, uint newOperationFee) public onlyOperator {  
    liquidityFee = newLiquidityFee;  
    operationFee = newOperationFee;  
}
```

Contract owner can exclude wallet from fees

```
function setExcludeFromFee(address account, bool exclude) public onlyOperator {  
    require(account != address(this), "Can't process the contract address.");  
    excludedFromFees[account] = exclude;  
    ExcludeAccountFromFee(account, exclude);  
}
```

Contract owner can set/change buy, sell, transfer tax

```
uint constant public MAX_BUY_TAX = 500;      // 5%  
uint constant public MAX_SELL_TAX = 500;      // 5%  
uint constant public MAX_TRANSFER_TAX = 0;    // 0%
```

```
function setTaxes(uint newBuyTax, uint newSellTax, uint newTransferTax) public onlyOperator {  
    buyTax = newBuyTax >= MAX_BUY_TAX ? MAX_BUY_TAX : newBuyTax;  
    sellTax = newSellTax >= MAX_SELL_TAX ? MAX_SELL_TAX : newSellTax;  
    transferTax = newTransferTax >= MAX_TRANSFER_TAX ? MAX_TRANSFER_TAX : newTransferTax;  
    emit OnSetTaxes(newBuyTax, newSellTax, newTransferTax);  
}
```

Contract owner can renounce ownership

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

Contract owner can exclude wallet from daily drop reward

```
function setExcludeFromDailyDrop(address account, bool exclude) public onlyOperator {  
    require(account != address(this), "Can't process the contract address.");  
    excludedFromDailyDrop[account] = exclude;  
    emit ExcludedAccountFromDailyDrop(account, exclude);  
}
```

Contract owner can exclude wallet from tx limits

```
function setExcludeFromMaxTokensPerWallet(address account, bool exclude) public onlyOperator {  
    require(account != address(this), "Can't process the contract address.");  
    excludedFromMaxTokensPerWallet[account] = exclude;  
    emit ExcludedAccountFromMaxTokensPerWallet(account, exclude);  
}
```

Contract owner can change wallet max tx limit

```
uint constant public MIN_TOKENS_PER_WALLET = 200; // 2%
```

```
function setMaxTokensPerWallet(uint256 newMaxTokensPerWallet) public onlyOperator {  
    require(newMaxTokensPerWallet >= MIN_TOKENS_PER_WALLET, "Can't set less than 2%");  
    _maxTokensPerWallet = TOTAL_SUPPLY * newMaxTokensPerWallet / 10000;  
    emit ChangedMaxTokensPerWallet(newMaxTokensPerWallet);  
}
```

Contract owner can change minimum tokens amount to liquify

```
function setMinTokensToLiquify(uint256 newMinTokensToLiquify) public onlyOperator {  
    minTokensToLiquify = newMinTokensToLiquify * 10**_decimals;  
}
```



CONCLUSION AND ANALYSIS

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.



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: 2.5%

Sell fees: 5%

Max TX: N/A

Max Sell: N/A

Honeypot Risk

Ownership: Owned

Blacklist: Not detected

Modify Max TX: Not detected

Modify Max Sell: Not detected

Disable Trading: Not detected

Rug Pull Risk

Liquidity: N/A

Holders: Clean



IRISH SHIBA COIN TOKEN DISTRIBUTION & TOP 10 TOKEN HOLDERS



Rank	Address	Quantity (Token)	Percentage
1	0x3b7edbdb1fdc1be99172601b2e2181b5bf0d2e31	763,014,000	98.2000%
2	UniCrypt: Token Vesting	13,937,049	1.7937%
3	0xaad594b5926a7d5fbc61985390baaf936a6b8d	48,951	0.0063%

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.

