freshcoins

SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS REPORT

Belicoin

$BLC

19/04/2024

TOKEN OVERVIEW

Fees

• Buy fees:	12%
• Sell fees:	12%
• Transfer fees:	12%

Fees privileges

- Can change buy fees up to 25%, sell fees up to 25% and transfer fees up to 25%

Ownership

- Owned

Minting

- No mint function

Max Tx Amount / Max Wallet Amount

- Can't change max tx amount and / or max wallet amount

Blacklist

- Blacklist function not detected

Other privileges

- Contract owner has to call enableTrading function to enable trade
- Can exclude / include from fees
- Can exclude / include from dividends

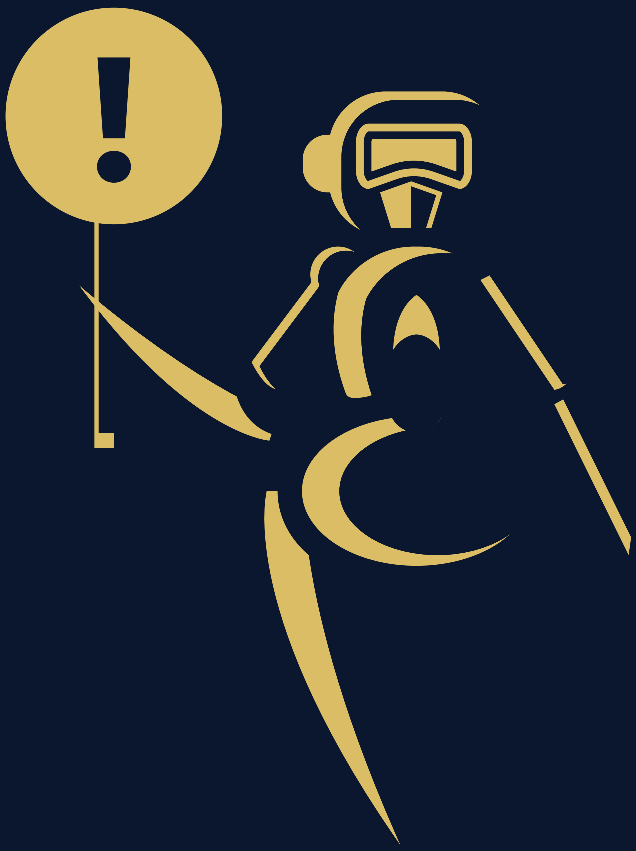
TABLE OF CONTENTS

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
Belicoin (Customer) to conduct a Smart Contract Code Review and Security Analysis.

0xC3Dc1F0eD23bd771dc8E92b5934aAd493F376fF0

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 19/04/2024

WEBSITE DIAGNOSTIC

https://belicoin.io/

🟥	🟨	🟩
0-49	50-89	90-100

86	91	94	92	NA
Performance	Accessibility	Best Practices	SEO	Progressive Web App

Socials

Twitter

https://twitter.com/BelicoinToken

Telegram

https://t.me/OfficialBelicoinSpanish

AUDIT OVERVIEW

82

Security Score

98

Static Scan
Automatic scanning for common vulnerabilities

80

ERC Scan
Automatic checks for ERC's conformance

1	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	Low
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't exclude addresses from transactions

🔴 Contract owner has to call enableTrading function to enable trade

Please note that any whitelisted wallet address retains the ability to engage in trading, even in situations where trading has been disabled

```
function enableTrading() external onlyOwner {
      require(!tradingEnabled, "EnableTrading: Trading was enabled already");
      tradingEnabled = true;

      emit TradingEnabled();
}

function _beforeTokenTransfer(address from, address to, uint256 amount)
      internal
      override
   {
      // Interactions with DEX are disallowed prior to enabling trading by owner
      if ((AMMPairs[from] && !isExcludedFromTradingRestriction[to]) || (AMMPairs[to] && !isExcludedFromTradingRestriction[from])) {
         require(tradingEnabled, "EnableTrading: Trading was not enabled yet");
      }

      super._beforeTokenTransfer(from, to, amount);
}
```

🟢 Contract owner can exclude/include wallet from trading restrictions (whitelist)

```
function excludeFromTradingRestriction(address account, bool isExcluded) public onlyOwner {
      isExcludedFromTradingRestriction[account] = isExcluded;

      emit ExcludeFromTradingRestriction(account, isExcluded);
}
```

🟢 Contract owner can exclude/include wallet from tax

```
function excludeFromFees(address account, bool isExcluded) public onlyOwner {
      isExcludedFromFees[account] = isExcluded;

      emit ExcludeFromFees(account, isExcluded);
}
```

Contract owner can exclude/include wallet from dividends

```
function excludeFromDividends(address account, bool isExcluded) external onlyOwner {
    _excludeFromDividends(account, isExcluded);
}

function _excludeFromDividends(address account, bool isExcluded) internal override {
    dividendTracker.excludeFromDividends(account, balanceOf(account), isExcluded);
}
```

Contract owner can change sell fees up to 25%, buy fees up to 25% and transfer fees up to 25%

```
function liquidityFeesSetup(uint16 _buyFee, uint16 _sellFee, uint16 _transferFee) public onlyOwner {
    totalFees[0] = totalFees[0] - liquidityFees[0] + _buyFee;
    totalFees[1] = totalFees[1] - liquidityFees[1] + _sellFee;
    totalFees[2] = totalFees[2] - liquidityFees[2] + _transferFee;
    require(totalFees[0] <= 2500 && totalFees[1] <= 2500 && totalFees[2] <= 2500, "TaxesDefaultRouter:
Cannot exceed max total fee of 25%");
    liquidityFees = [_buyFee, _sellFee, _transferFee];
    emit liquidityFeesUpdated(_buyFee, _sellFee, _transferFee);
}

function rewardsFeesSetup(uint16 _buyFee, uint16 _sellFee, uint16 _transferFee) public onlyOwner {
    totalFees[0] = totalFees[0] - rewardsFees[0] + _buyFee;
    totalFees[1] = totalFees[1] - rewardsFees[1] + _sellFee;
    totalFees[2] = totalFees[2] - rewardsFees[2] + _transferFee;
    require(totalFees[0] <= 2500 && totalFees[1] <= 2500 && totalFees[2] <= 2500, "TaxesDefaultRouter:
Cannot exceed max total fee of 25%");
    rewardsFees = [_buyFee, _sellFee, _transferFee];
    emit rewardsFeesUpdated(_buyFee, _sellFee, _transferFee);
}

function autoBurnFeesSetup(uint16 _buyFee, uint16 _sellFee, uint16 _transferFee) public onlyOwner {
    totalFees[0] = totalFees[0] - autoBurnFees[0] + _buyFee;
    totalFees[1] = totalFees[1] - autoBurnFees[1] + _sellFee;
    totalFees[2] = totalFees[2] - autoBurnFees[2] + _transferFee;
    require(totalFees[0] <= 2500 && totalFees[1] <= 2500 && totalFees[2] <= 2500, "TaxesDefaultRouter:
Cannot exceed max total fee of 25%");
    autoBurnFees = [_buyFee, _sellFee, _transferFee];
    emit autoBurnFeesUpdated(_buyFee, _sellFee, _transferFee);
}
```

Contract owner can change swap settings

```
function updateSwapThreshold(uint16 _swapThresholdRatio) public onlyOwner {
    require(_swapThresholdRatio > 0 && _swapThresholdRatio <= 500, "SwapThreshold: Cannot exceed limits
from 0.01% to 5% for new swap threshold");
    swapThresholdRatio = _swapThresholdRatio;

    emit SwapThresholdUpdated(_swapThresholdRatio);
}
```

Contract owner can transfer ownership

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

function _transferOwnership(address newOwner) internal virtual {
    address oldOwner = _owner;
    _owner = newOwner;
    emit OwnershipTransferred(oldOwner, newOwner);
}
```

Contract owner can renounce ownership

```
function renounceOwnership() public virtual onlyOwner {
    _transferOwnership(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 1 HIGH issues during the first review.

TOKEN DETAILS

Details

Buy fees:	12%
Sell fees:	12%
Transfer fees:	12%
Max TX:	N/A
Max Wallet:	N/A

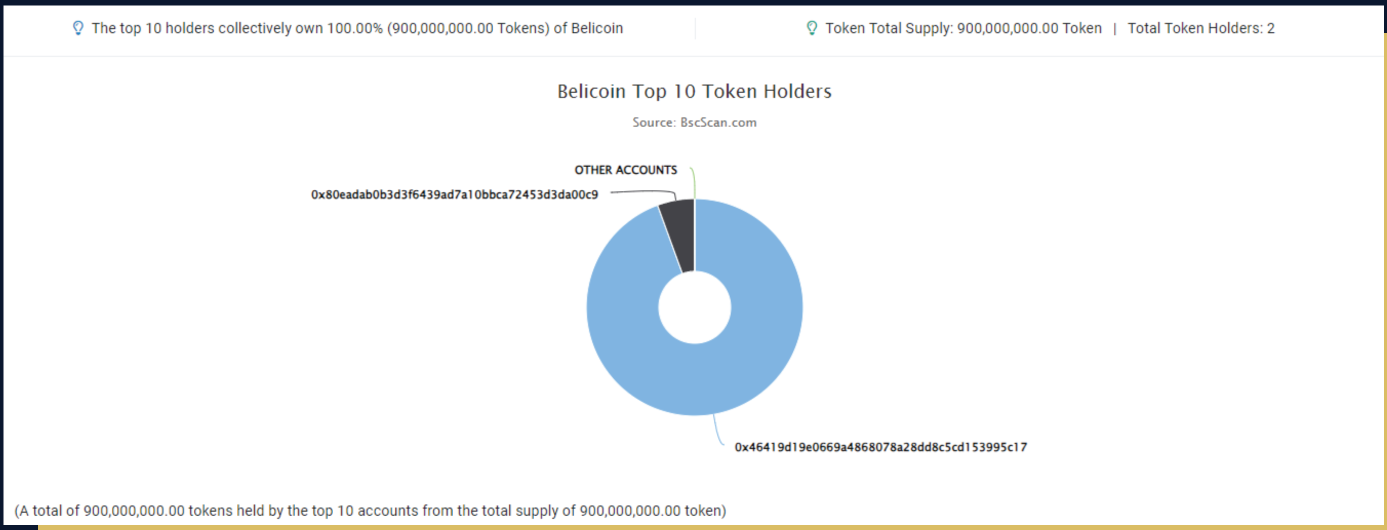
Honeypot Risk

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:	100% unlocked tokens

BLC TOKEN ANALYTICS
& TOP 10 TOKEN HOLDERS



The top 10 holders collectively own 100.00% (900,000,000.00 Tokens) of Belicoin

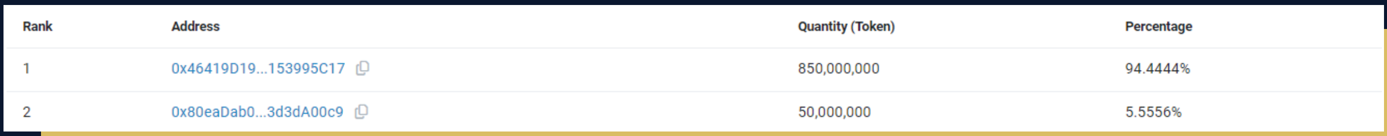
Token Total Supply: 900,000,000.00 Token | Total Token Holders: 2

Belicoin Top 10 Token Holders

Source: BscScan.com

OTHER ACCOUNTS
0x80eadab0b3d3f6439ad7a10bbca72453d3da00c9
0x46419d19e0669a4868078a28dd8c5cd153995c17

(A total of 900,000,000.00 tokens held by the top 10 accounts from the total supply of 900,000,000.00 token)



Rank	Address	Quantity (Token)	Percentage
1	0x46419D19...153995C17	850,000,000	94.4444%
2	0x80eaDab0...3d3dA00c9	50,000,000	5.5556%

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.