freshcoins

SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS REPORT

FIFABET2022

$FB22

13/11/2022

TOKEN OVERVIEW

Fees

- Buy fees: 5%
- Sell fees: 5%

Fees privileges

- Can’t set / change fees

Ownership

- Owned

Minting

- No mint function

Max Tx Amount / Max Wallet Amount

- Can set max tx amount and / or wallet limitations with threshold

Blacklist

- No blacklist function

Other privileges

- Can exclude from fees

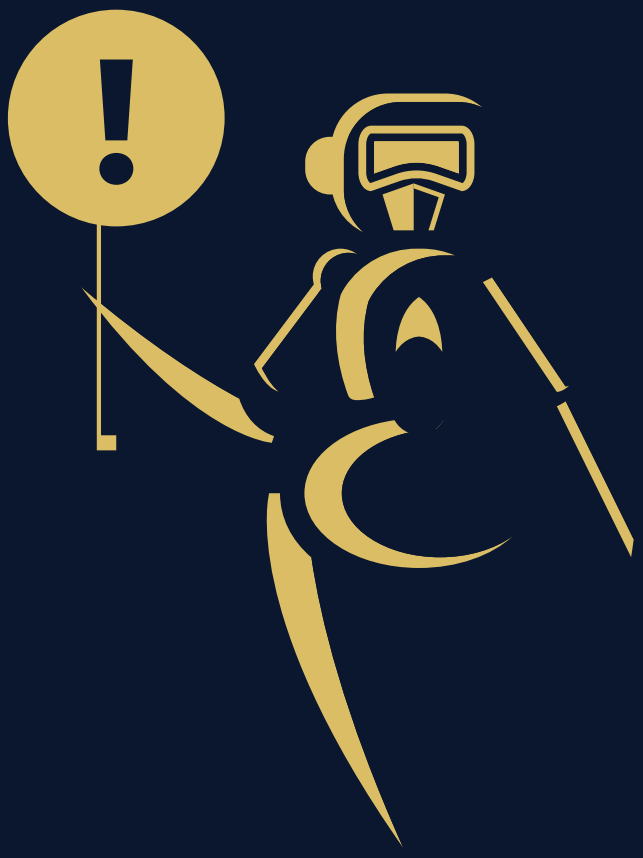
TABLE OF CONTENTS

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 FIFABET2022 (Customer) to conduct a Smart Contract Code Review and Security Analysis.

0xF1FA20228a94C305bF4cD55a765636B3665d4D20

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 13/11/2022

AUDIT OVERVIEW

86

Security Score

90

Static Scan
Automatic scanning for common vulnerabilities

82

ERC Scan
Automatic checks for ERC's conformance

0 High

1 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	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't exclude an address from transactions

- Contract owner can exclude/include wallet(s) from fees

```
function updateExemptFee(address _address, bool state) external onlyOwner {
        exemptFee[_address] = state;
}

function bulkExemptFee(address[] memory accounts, bool state) external onlyOwner {
        for (uint256 i = 0; i < accounts.length; i++) {
            exemptFee[accounts[i]] = state;
        }
}
```

- Contract owner has to call `EnableTrading()` function to open trade

Can't be disabled once enabled.

Current value: tradingEnabled: False (bool)

```
function EnableTrading(uint256 _deadblock) external onlyOwner {
        require(!tradingEnabled, "Cannot re-enable trading");
        require(_deadblock < 5,"Deadblock should be less than 5 Blocks");
        deadblock = _deadblock;
        tradingEnabled = true;
        providingLiquidity = true;
        genesis_block = block.number;
}
```

- Contract owner can change max buy limitations, max sell limitations and max wallet limitations

Current values:

maxBuyLimit: 100000000000000000000000 **uint256 (100.000.000 FB22)**

maxSellLimit: 100000000000000000000000 **uint256 (100.000.000 FB22)**

maxWalletLimit: 100000000000000000000000 **uint256 (100.000.000 FB22)**

```
function updateMaxTxLimit(uint256 maxBuy, uint256 maxSell, uint256 maxWallet) external onlyOwner {
        require(maxBuy >= 100_000, "Cannot set max buy amount lower than 0.1%");
        require(maxSell >= 100_000, "Cannot set max sell amount lower than 0.1%");
        require(maxWallet >= 1_000_000, "Cannot set max wallet amount lower than 1%");
        maxBuyLimit = maxBuy * 10**decimals();
        maxSellLimit = maxSell * 10**decimals();
        maxWalletLimit = maxWallet * 10**decimals();
}
```

Contract owner can change devWallet and marketingWallet addresses

Current values:

devWallet : 0x3fb60c4449dce705d0b8bde716ec0a79dd132603

marketingWallet : 0xff3a276e63b46b11e75674afe6fc31ceeed87f3e

```
function updateMarketingWallet(address newWallet) external onlyOwner {
    require(newWallet != address(0),"Fee Address cannot be zero address");
    marketingWallet = newWallet;
}

function updateDevWallet(address newWallet) external onlyOwner {
    require(newWallet != address(0),"Fee Address cannot be zero address");
    devWallet = newWallet;
}
```

Contract owner can change swap settings

```
function updateLiquidityProvide(bool state) external onlyOwner {
    //update liquidity providing state
    providingLiquidity = state;
}

function updateLiquidityTreshhold(uint256 new_amount) external onlyOwner {
    //update the treshhold
    require(new_amount <= 1_000_000, "Swap threshold amount should be lower or euqal to 1% of
tokens");
    tokenLiquidityThreshold = new_amount * 10**decimals();
}
```

Contract owner can withdraw stuck BNB or tokens ($FB22 excluded)

```
function rescueBNB(uint256 weiAmount) external onlyOwner {
    payable(owner()).transfer(weiAmount);
}

function rescueBSC20(address tokenAdd, uint256 amount) external onlyOwner {
    require(tokenAdd != address(this), "Owner can't claim contract's balance of its own tokens");
    IBEP20(tokenAdd).transfer(owner(), amount);
}
```

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);
}

function _setOwner(address newOwner) private {
    address oldOwner = _owner;
    _owner = newOwner;
    emit OwnershipTransferred(oldOwner, newOwner);
}
```

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 HIGH issues during the first review.

TOKEN DETAILS

Details

Buy fees:	5%
Sell fees:	5%
Max TX:	100,000,000
Max Sell:	100,000,000

Honeypot Risk

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

Others

Liquidity:	N/A
Holders:	Clean

FB22 TOKEN ANALYTICS & TOP 10 TOKEN HOLDERS

The top 10 holders collectively own 100.00% (100,000,000.00 Tokens) of FIFABET2022

Token Total Supply: 100,000,000.00 Token | Total Token Holders: 1

(A total of 100,000,000.00 tokens held by the top 10 accounts from the total supply of 100,000,000.00 token)

Rank	Address	Quantity (Token)	Percentage
1	0xebb8f71145f96aa4d932897ce5e1b0d777140f7d	100,000,000	100.0000%

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.