freshcoins

SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS REPORT

TRUMP ICE

$TICE

07/04/2024

TOKEN OVERVIEW

Fees

- Buy fees: 5%
- Sell fees: 5%
- Transfer fees: 0%

Fees privileges

- Can’t change / set fees

Ownership

- Owned

Minting

- No mint function

Max Tx Amount / Max Wallet Amount

- Can’t change max tx amount or wallet amount

Blacklist

- No blacklist function

Other privileges

- Contract owner has the ability to update the fund wallet without requiring any additional checks

TABLE OF CONTENTS

1 DISCLAIMER

2 INTRODUCTION

3 WEBSITE + SOCIALS

4-5 AUDIT OVERVIEW

6-8 OWNER PRIVILEGES

9 CONCLUSION AND ANALYSIS

10 TOKEN DETAILS

11 TICE TOKEN ANALYTICS &
TOP 10 TOKEN HOLDERS

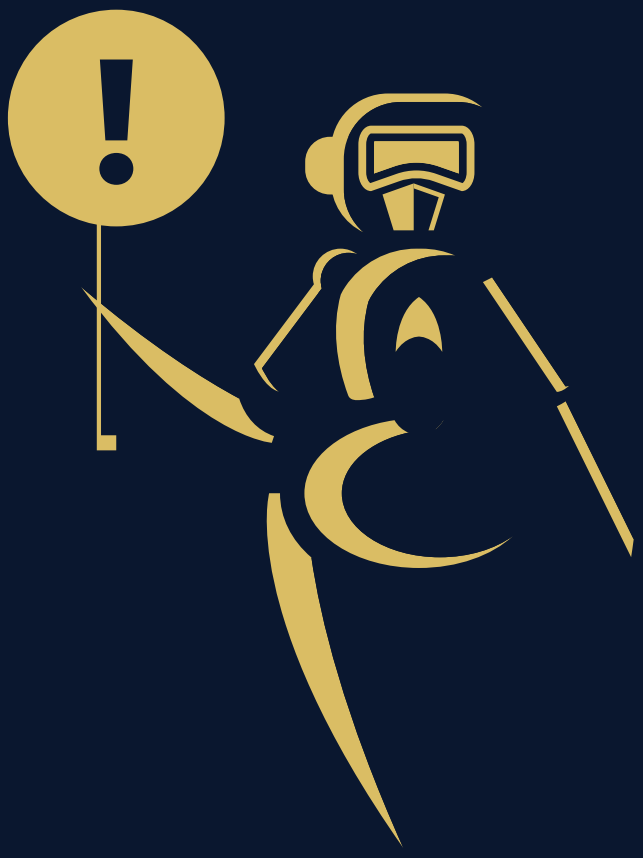
12 TECHNICAL DISCLAIMER

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

0xB123CcBE05f4a301509bf2cE6BFB1eC8f0E57ba3

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 07/04/2024

WEBSITE DIAGNOSTIC

https://trumpice.xyz/


0-49


50-89


90-100

Performance

Accessibility

Best Practices

SEO

Progressive Web App

Socials


Twitter

http://x.com/trumpice_


Telegram

https://t.me/trumpice

AUDIT OVERVIEW

55

Security Score
HIGH RISK
Audit FAIL

93

Static Scan
Automatic scanning for common vulnerabilities

59

ERC Scan
Automatic checks for ERC's conformance

1	High
2	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	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 disable trading.

- Contract owner can change fund wallet address

Since the function lacks additional checks, the contract owner could potentially set the fund wallet to a contract address that is unable to receive ETH/BNB, leading to the transformation of the token contract into a honeypot. This scenario would render selling impossible for users

Default value: 0xdf949410F4Ac63270b7eacb0BDf844c3a2BFD8eD

```
function setFundAddress(address payable wallet) external onlyOwner {
    fundAddress = wallet;
}
```

The smart contract was deployed throught the fatsale.finance website

Was listed several settings associated with this deployment:

enableChangeTax: False

enableKillBlock: False

enableOffTrade: False

enableRewardList: False

enableSwapLimit: False

enableTransferFee: False

enableWalletLimit: False

The functions associated with these settings are not available in the smart contract. This means that despite these settings being applied during deployment, the functionalities they represent might not be implemented or accessible within the contract code. Therefore, users interacting with this contract may not be able to utilize features such as enable/disable trade, or change tax, among others, depending on the specific settings and associated functions

Contract owner can enable/disable transfer fee and set up to 25%

```
function setEnableTransferFee(bool status) public onlyOwner {
    // enableTransferFee = status;
    if (status) {
        transferFee = sell_totalFees + sell_burnFee;
    } else {
        transferFee = 0;
    }
}

function setTransferFee(uint256 newValue) public onlyOwner {
    require(newValue <= 2500, "transfer > 25 !");
    transferFee = newValue;
}
```

Contract owner can change swap settings (without threshold)

```
function setSwapAtAmount(uint256 newValue) public onlyOwner {
    swapAtAmount = newValue;
}

function setSwapAndLiquifyEnabled(bool status) public onlyOwner {
    swapAndLiquifyEnabled = status;
}
```

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:	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

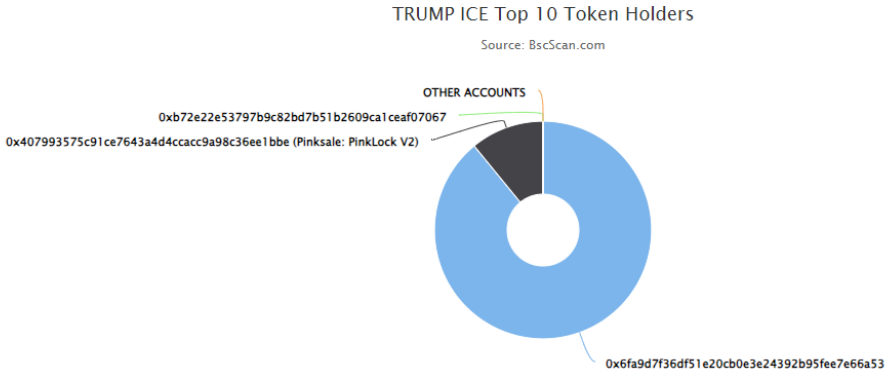
Others

Liquidity:	N/A
Holders:	Clean

TICE TOKEN ANALYTICS & TOP 10 TOKEN HOLDERS

The top 10 holders collectively own 100.00% (10,000,000,000.00 Tokens) of TRUMP ICE

Token Total Supply: 10,000,000,000.00 Token | Total Token Holders: 3



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

Rank	Address	Quantity (Token)	Percentage
1	0x6fA9d7F3...EE7e66A53	8,907,000,000	89.0700%
2	Pinksale: PinkLock V2	1,092,000,000	10.9200%
3	0xb72E22E5...cEaF07067	1,000,000	0.0100%

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.