freshcoins

SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS REPORT

Tom coin

$TMC

07/04/2022

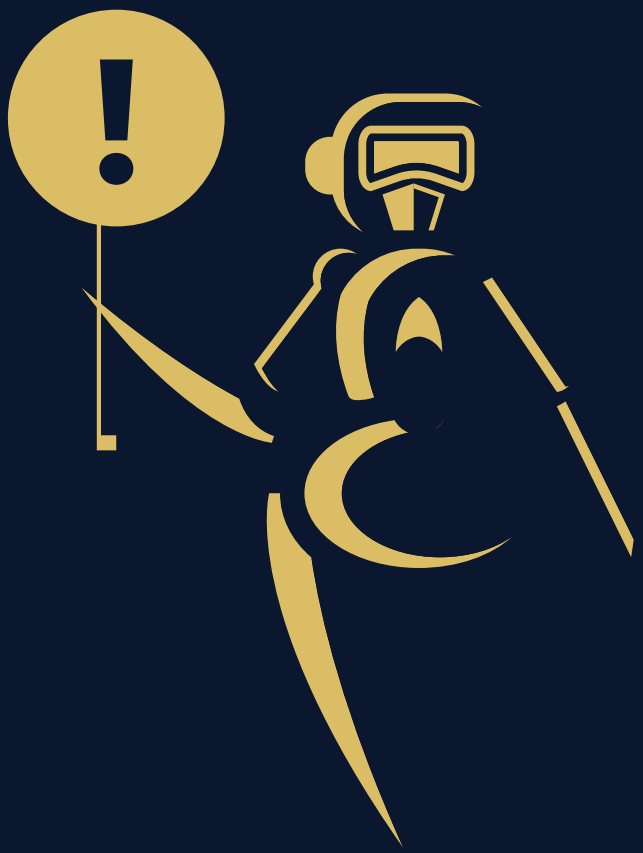
TABLE OF CONTENTS

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

0xb27e4DE03A6823Ddcf4e53606D5AeEC295377D70

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 07/04/2022

AUDIT OVERVIEW

98

Security Score

98

Static Scan
Automatic scanning for common vulnerabilities

98

ERC Scan
Automatic checks for ERC's conformance

0 High

0 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 exclude an address from transactions

Contract owner can mint tokens after initial contract deploy

```
function mint(address user, uint256 value) external onlyOwner {
    require(isMintable, "NOT_MINTABEL_TOKEN");
    require(user == _OWNER_, "NOT_OWNER");

    balances[user] = balances[user].add(value);
    totalSupply = totalSupply.add(value);
    emit Mint(user, value);
    emit Transfer(address(0), user, value);
}
```

Contract owner can burn tokens

```
function burn(uint256 value) external {
    require(isMintable, "NOT_MINTABEL_TOKEN");
    require(balances[msg.sender] >= value, "VALUE_NOT_ENOUGH");

    balances[msg.sender] = balances[msg.sender].sub(value);
    totalSupply = totalSupply.sub(value);
    emit Burn(msg.sender, value);
    emit Transfer(msg.sender, address(0), value);
}
```

Contract owner can change team address

Current value:

team : 0xea022dd6265110d640f6f916c0c1e38c51b93c78

```
function changeTeamAccount(address newTeam) external onlyOwner {
    require(tradeFeeRatio > 0, "NOT_TRADE_FEE_TOKEN");
    emit ChangeTeam(team,newTeam);
    team = newTeam;
}
```

Contract owner can transfer ownership

```
function transferOwnership(address newOwner) public onlyOwner {
    emit OwnershipTransferPrepared(_OWNER_, newOwner);
    _NEW_OWNER_ = newOwner;
}
```

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 issue during the first review.

TOKEN DETAILS

Details

Buy fees:	3%
Sell fees:	3%
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

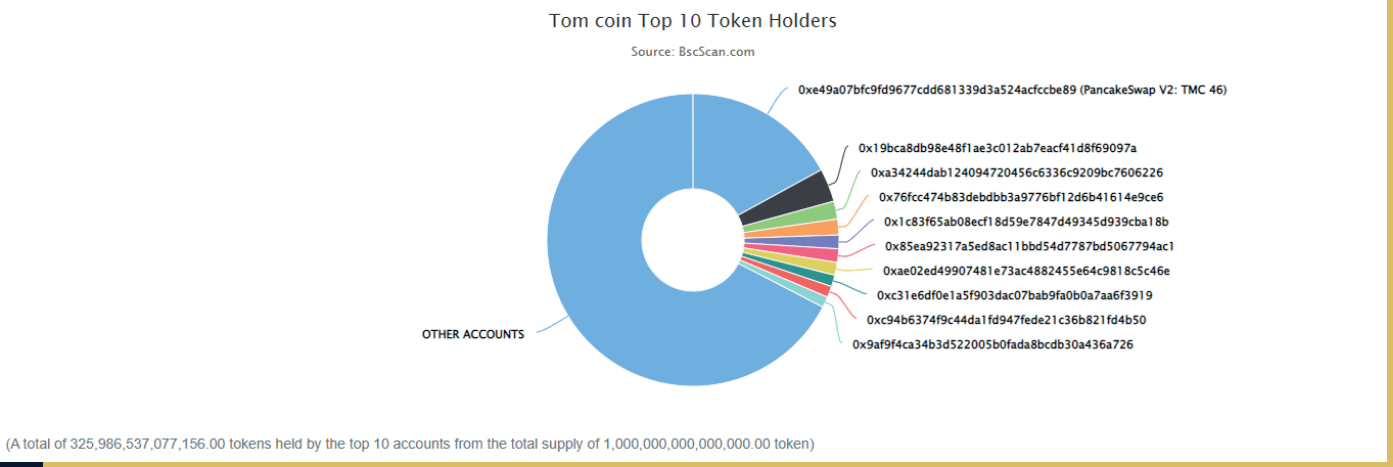
Rug Pull Risk

Liquidity:	N/A
Holders:	Clean

TMC TOKEN ANALYTICS & TOP 10 TOKEN HOLDERS

The top 10 holders collectively own 32.60% (325,986,537,077,156.00 Tokens) of Tom coin

Token Total Supply: 1,000,000,000,000,000.00 Token | Total Token Holders: 1,151



(A total of 325,986,537,077,156.00 tokens held by the top 10 accounts from the total supply of 1,000,000,000,000,000.00 token)

Rank	Address	Quantity (Token)	Percentage
1	PancakeSwap V2: TMC 46	170,417,748,009,254.169643467	17.0418%
2	0x19bca8db98e48f1ae3c012ab7eacf41d8f69097a	36,930,976,868,136.47349732	3.6931%
3	0xa34244dab124094720456c6336c9209bc7606226	19,748,302,881,510.921189172	1.9748%
4	0x76fcc474b83debdbb3a9776bf12d6b41614e9ce6	17,507,454,872,699.106157382	1.7507%
5	0x1c83f65ab08ecf18d59e7847d49345d939cba18b	15,090,236,204,311.653872025	1.5090%
6	0x85ea92317a5ed8ac11bbd54d7787bd5067794ac1	14,938,032,499,153.222839037	1.4938%
7	0xae02ed49907481e73ac4882455e64c9818c5c46e	14,200,000,000,000	1.4200%
8	0xc31e6df0e1a5f903dac07bab9fa0b0a7aa6f3919	12,690,257,402,512.834014194	1.2690%
9	0xc94b6374f9c44da1fd947fede21c36b821fd4b50	12,396,559,974,653.481756264	1.2397%
10	0x9af9f4ca34b3d522005b0fada8bcdb30a436a726	12,066,968,364,924.528	1.2067%

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.