freshcoins

SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS REPORT

Rich Quack 2.0

$RICHQUACK20

23/07/2023

TOKEN OVERVIEW

Fees

- Buy fees: N/A
- Sell fees: N/A

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

- N/A

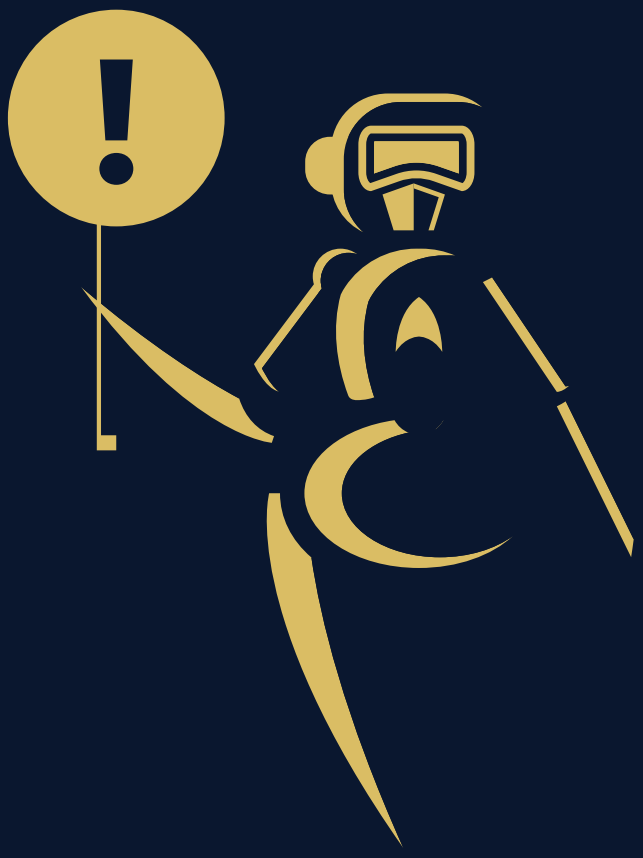
TABLE OF CONTENTS

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
Rich Quack 2.0 (Customer) to conduct a Smart Contract Code Review and Security Analysis.

0x917fA17B0508aC8366daDD429B7934774A393dBa

Network: Ethereum (ETH)

This report presents the findings of the security assessment of Customer's smart contract and its code review conducted on 23/07/2023

WEBSITE DIAGNOSTIC

https://richquack20.vip/


0-49


50-89


90-100

Performance

Accessibility

Best Practices

SEO

Progressive Web App

Socials


Twitter

https://twitter.com/richquack2eth


Telegram

https://t.me/richquack2eth

AUDIT OVERVIEW

72

Security Score
HIGH RISK
Audit FAIL

93

Static Scan
Automatic scanning for common vulnerabilities

64

ERC Scan
Automatic checks for ERC's conformance

1 High

1 Medium

0 Low

1 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	Low
6	Reverted DoS	Passed
7	DoS with block gas limit	Low
8	Methods execution permissions	Passed
9	Exchange rate impact	Passed
10	Malicious Event	Low
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't exclude an address from transactions.
- Contract owner can't set / change buy & sell taxes.
- Contract owner can’t change swap settings.
- Contract owner can’t change tx amount
- Use latest version of the solidity

  The contract uses Solidity version 0.4.24, which is relatively old
- ApproveAndCallFallBack interface is used in the RichQuack20 contract through the approveAndCall function, enabling other contracts to receive token approval and invoke specific functions within RichQuack20 in a single transaction.

  The approveAndCall function allows a token holder to approve another contract (spender) to spend a certain amount of tokens on their behalf. This introduces the risk of approving malicious or vulnerable contracts that may misuse the approved token amount. Users should exercise caution and only approve contracts they trust.

  The approveAndCall function does not perform any validation or checks on the data parameter being passed to the receiveApproval function in the spender contract. Depending on how the receiveApproval function is implemented in the spender contract, it could potentially lead to unintended behavior if the data is not validated properly.

  The receiveApproval function in the ApproveAndCallFallBack interface is called directly after approving the spender contract. If the spender contract contains any reentrant calls or if it interacts with other contracts that have reentrancy vulnerabilities, it could result in unexpected behavior and potentially lead to funds being drained.

  The contract owner could approve a malicious or vulnerable contract as the spender, giving it access to spend tokens. If the malicious contract has not been thoroughly audited or contains security vulnerabilities, it could misuse the approved tokens to perform unauthorized actions or exploit the contract.

```
contract ApproveAndCallFallBack {
  function receiveApproval(address from, uint256 tokens, address token, bytes data) public;
}

function approveAndCall(address spender, uint tokens, bytes data) public returns (bool success) {
    allowed[msg.sender][spender] = tokens;
    emit Approval(msg.sender, spender, tokens);
    ApproveAndCallFallBack(spender).receiveApproval(msg.sender, tokens, this, data);
    return true;
}
```

Contract owner can burn tokens

```
function burn(uint amount) external {
      balances[msg.sender] -= amount;
      _totalSupply -= amount;
      emit Transfer(msg.sender, address(0), amount);
}
```

Contract owner can renounce ownership

```
function renounceOwnership() public onlyOwner {
  emit OwnershipRenounced(owner);
  owner = 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:	N/A
Sell fees:	N/A
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

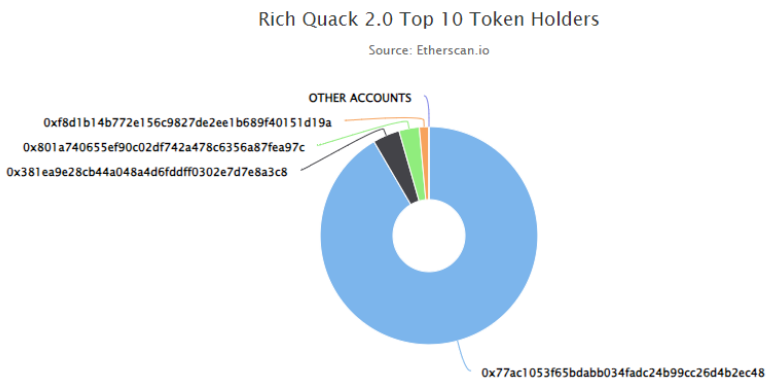
Others

Liquidity:	N/A
Holders:	Clean

RICHQUACK20 TOKEN ANALYTICS & TOP 10 TOKEN HOLDERS

The top 10 holders collectively own 100.00% (1,000,000,000,000,000.00 Tokens) of Rich Quack 2.0

Token Total Supply: 1,000,000,000,000,000.00 Token | Total Token Holders: 4



(A total of 1,000,000,000,000,000.00 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	0x77ac10...D4B2ec48	915,750,000,000,000	91.5750%
2	0x381Ea9...D7e8A3c8	40,000,000,000,000	4.0000%
3	0x801a74...87Fea97C	30,000,000,000,000	3.0000%
4	0xf8D1b1...0151D19A	14,250,000,000,000	1.4250%

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.