



## SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS REPORT



**SpCard**

\$SpCard



11/01/2022



# TABLE OF CONTENTS

- 1 DISCLAIMER
- 2 INTRODUCTION
- 3-4 WEBSITE DIAGNOSTIC
- 5-6 AUDIT OVERVIEW
- 7 OWNER PRIVILEGES
- 8 CONCLUSION AND ANALYSIS
- 9 TOKEN DETAILS
- 10 SPCARD TOKEN DISTRIBUTION & TOP 10 TOKEN HOLDERS
- 11 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 SpCard (Customer) to conduct a Smart Contract Code Review and Security Analysis.

0xd026645B685DdbFA838683800C11483B4B396ce0

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 11/01/2022



# WEBSITE DIAGNOSTIC

<https://spcard.finance/>



0-49



50-89



90-100



Performance



Accessibility



Best Practices



SEO



Progressive  
Web App

## Metrics



First Contentful Paint

**4.3 s**



Time to interactive

**9.1 s**



Speed Index

**5.2 s**



Total Blocking Time

**720 ms**



Large Contentful Paint

**13.8 s**



Cumulative Layout Shift

**0.102**

# WEBSITE IMPROVEMENTS

---

Serve images in next-gen formats

---

Properly size images

---

Reduce unused JavaScript

---

Reduce unused CSS

---

Ensure text remains visible during webfont load

---

Reduce the impact of third-party code Third-party code blocked the main thread for 420 ms

---

Image elements do not have explicit `width` and `height`

---

Reduce JavaScript execution time - 2.0 s

---

`<html>` element does not have a `[lang]` attribute

---

# AUDIT OVERVIEW



**Security Score**



**Static Scan**  
Automatic scanning for common vulnerabilities



**ERC Scan**  
Automatic checks for ERC's conformance

0 High

0 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	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 an address from transactions.**

**Contract owner can exclude/include wallet from fee**

```
function excludeFromFee(address account) external onlyOwner {
    _isExcludedFromFee[account] = true;
    emit ExcludedFromFee(account);
}

function includeInFee(address account) external onlyOwner {
    _isExcludedFromFee[account] = false;
    emit IncludedToFee(account);
}
```

**Contract owner can renounce ownership**

```
function renounceOwnership() public virtual onlyOwner {
    emit OwnershipTransferred(_owner, address(0));
    _owner = 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");
    emit OwnershipTransferred(_owner, newOwner);
    _owner = newOwner;
}
```

**Contract owner can change the fees**

```
function setFees(uint256 bnbFee, uint256 marketingFee) external onlyOwner() {
    require(_marketingFee != marketingFee && marketingFee <= 5);
    _marketingFee = marketingFee;
    emit UpdateFees(bnbFee, marketingFee);
}
```

# 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 LOW issue 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

## Rug Pull Risk

Liquidity: 19% Locked

Holders: Clean



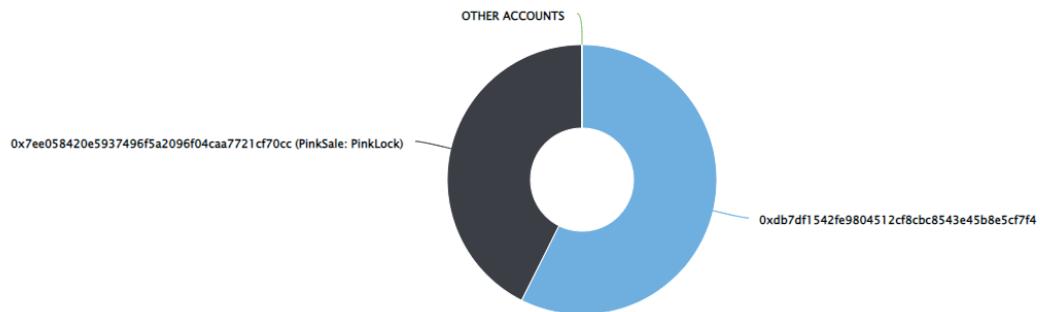
# SPCARD TOKEN DISTRIBUTION & TOP 10 TOKEN HOLDERS

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

Token Total Supply: 10,000,000,000.00 Token | Total Token Holders: 2

SpCard Top 100 Token Holders

Source: BscScan.com



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

Rank	Address	Quantity (Token)	Percentage
1	0xdb7df1542fe9804512cf8cbc8543e45b8e5cf7f4	5,740,000,000	57.4000%
2	PinkSale: PinkLock	4,260,000,000	42.6000%

# 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.

