



SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS REPORT



GLOBALPad
\$GLOBAL

09/03/2022

TABLE OF CONTENTS

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

0x9b199C587a930b653a95316352aeD623D5306E18

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 09/03/2022



AUDIT OVERVIEW



Security Score



Static Scan
Automatic scanning for common vulnerabilities



ERC Scan
Automatic checks for ERC's conformance



High



Medium



Low



Optimizations



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't mint tokens after initial contract deploy

Contract owner can exclude/include wallet from tax

```
function excludeFromFees(address account, bool excluded) public onlyOwner {  
    _isExcludedFromFees[account] = excluded;  
    emit ExcludeFromFees(account, excluded);  
}
```

Contract owner can change the fees up to 20%

```
function BuyFees( uint256 _dev, uint256 _marketing, uint256 _liquidity) external onlyOwner {  
  
    buyDevFee = _dev;  
    require(buyDevFee <= 2, "fees upto 2%");  
  
    buyMarketingFee = _marketing;  
    require(buyMarketingFee <= 2, "fees upto 2%");  
  
    buyLiquidityFee = _liquidity;  
    require(buyLiquidityFee <= 1, "fees upto 1%");  
  
    buyTotalFees = buyMarketingFee + buyLiquidityFee + buyDevFee;  
    require(buyTotalFees <= 5, "Buy fees upto 5%");  
  
}  
  
function SellFees(uint256 _dev, uint256 _marketing, uint256 _liquidity) external onlyOwner {  
  
    sellDevFee = _dev;  
    require(sellLiquidityFee <= 6, "fees upto 6%");  
  
    sellMarketingFee = _marketing;  
    require(sellMarketingFee <= 7, "fees upto 7%");  
  
    sellLiquidityFee = _liquidity;  
    require(sellLiquidityFee <= 2, "fees upto 2%");  
  
    sellTotalFees = sellMarketingFee + sellLiquidityFee + sellDevFee;  
    require(sellTotalFees <= 15, "Sell fees upto 15%");  
  
}
```

Contract owner can change swap settings

```
function updateTaxTokenMinAmount(uint256 newAmount) external onlyOwner returns (bool){  
    require(newAmount >= totalSupply() * 1 / 100000, "Swap amount cannot be lower than 0.001% total supply.");  
    require(newAmount <= totalSupply() * 5 / 1000, "Swap amount cannot be higher than 0.5% total supply.");  
    TaxTokenMinAmount = newAmount;  
    return true;  
}
```

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;  
}
```



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:	0%
Sell fees:	10%
Max TX:	N/A
Max Sell:	N/A

Honeypot Risk

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

Rug Pull Risk

Liquidity:	N/A
Holders:	Clean



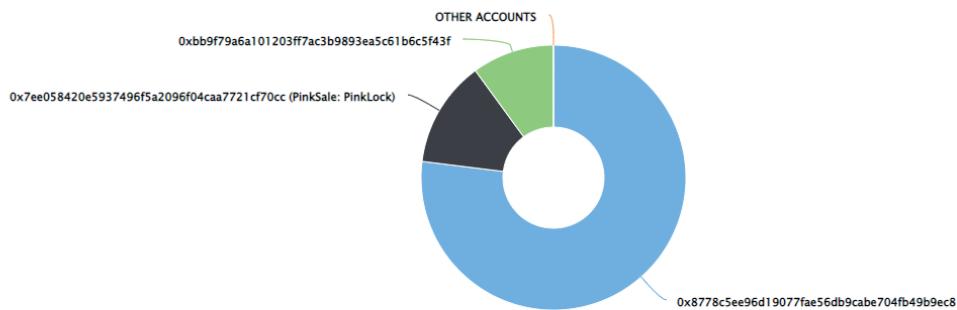
GLOBALPAD TOKEN ANALYTICS & TOP 10 TOKEN HOLDERS

The top 10 holders collectively own 100.00% (50,000,000.00 Tokens) of GLOBALPad

Token Total Supply: 50,000,000.00 Token | Total Token Holders: 3

GLOBALPad Top 10 Token Holders

Source: BscScan.com



(A total of 50,000,000.00 tokens held by the top 10 accounts from the total supply of 50,000,000.00 token)

Rank	Address	Quantity (Token)	Percentage
1	0x8778c5ee96d19077fae56db9cabe704fb49b9ec8	38,484,000	76.9680%
2	PinkSale: PinkLock	6,500,000	13.0000%
3	0xbb9f79a6a101203ff7ac3b9893ea5c61b6c5f43f	5,016,000	10.0320%

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.

