



SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS REPORT



GREAT DANE AI
\$GREATAI

24/03/2023



TOKEN OVERVIEW

Fees

- Buy fees: 1%
- Sell fees: 1%

Fees privileges

- Can change fees up to 10%

Ownership

- Owned

Minting

- No mint function

Max Tx Amount / Max Wallet Amount

- Can change max tx amount (with threshold)

Blacklist

- Blacklist function detected

Other privileges

- Can exclude / include from fees
-

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

GREATAI 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 **GREAT DANE AI** (Customer) to conduct a Smart Contract Code Review and Security Analysis.

0x18f01f43f3aa33833f4ec8a9b9218d862ba5d8e8

Network: **Polygon (Matic)**

This report presents the findings of the security assessment of Customer's smart contract and its code review conducted on **24/03/2023**



WEBSITE DIAGNOSTIC

<https://greatdaneai.com/>



0-49



50-89



90-100



Performance



Accessibility



Best
Practices



SEO



Progressive
Web App

Socials



Twitter

<https://twitter.com/greatdaneinu21>



Telegram

<https://t.me/+rvUcD2omBNswNzFh>

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 mint tokens after initial contract deploy

- Contract owner can exclude an address from transactions

```
function blacklistAccount(address account) external onlyOwner {
    isBlackListed[account] = true;
}

function unBlackListAccount(address account) external onlyOwner {
    isBlackListed[account] = false;
}
```

- Contract owner can exclude/include wallet from tax

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

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

- Contract owner can exclude/include wallet from rewards

```
function excludeFromReward(address account) public onlyOwner() {
    require(!_isExcluded[account], "Account is already excluded");
    if(_rOwned[account] > 0) {
        _tOwned[account] = tokenFromReflection(_rOwned[account]);
    }
    _isExcluded[account] = true;
    _excluded.push(account);
}

function includeInReward(address account) external onlyOwner() {
    require(_isExcluded[account], "Account is already excluded");
    for (uint256 i = 0; i < _excluded.length; i++) {
        if (_excluded[i] == account) {
            _excluded[i] = _excluded[_excluded.length - 1];
            _tOwned[account] = 0;
            _isExcluded[account] = false;
            _excluded.pop();
            break;
        }
    }
}
```

● Contract owner can change swap settings

```
function setSwapAndLiquifyEnabled(bool _enabled) public onlyOwner {
    swapAndLiquifyEnabled = _enabled;
    emit SwapAndLiquifyEnabledUpdated(_enabled);
}
```

● Contract owner can set max tx limitations (with threshold)

```
function setMaxTxAmount(uint256 maxTxAmount) external onlyOwner() {
    _maxTxAmount = maxTxAmount;
    require(_maxTxAmount > totalSupply().div(400), "value too low");
}
```

● Contract owner can change marketingWallet address

Current value:

marketingWallet : 0x879c193f0b8a21924ab274f5c1883affbfd7962a

```
function setMarketingWallet(address payable newWallet) external onlyOwner() {
    marketingWallet = newWallet;
}
```

● Contract owner can change fees up to 10%

```
function setFeePercent(uint256 taxFee, uint256 liquidityFee, uint256 marketingFee, uint256 burnFee) external onlyOwner() {
    require(taxFee.add(liquidityFee).add(marketingFee).add(burnFee) <= 10, "tax too high");
    _taxFee = taxFee;
    _liquidityFee = liquidityFee;
    _marketingFee = marketingFee;
    _burnFee = burnFee;
}
```

● 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;
}
```

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:	1%
Sell fees:	1%
Max TX:	100,000,000,000,000
Max Sell:	N/A

Honeypot Risk

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

Rug Pull Risk

Liquidity:	N/A
Holders:	Clean



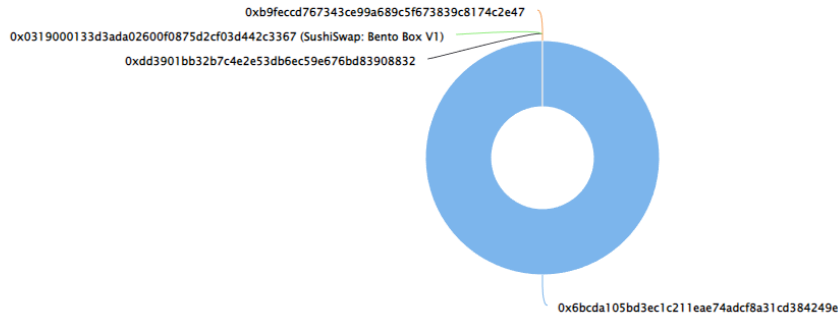
GREATAI TOKEN ANALYTICS & TOP 10 TOKEN HOLDERS

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

Token Total Supply: 100,000,000,000,000.00 Token | Total Token Holders: 6

GREAT DANE AI Top 10 Token Holders

Source: polygonscan.com



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

Rank	Address	Quantity (Token)	Percentage
1	0x6bcd105bd3ec1c211eae74adcf8a31cd384249e	99,999,995,899,045.576526537896435716	100.0000%
2	0xdd3901bb32b7c4e2e53db6ec59e676bd83908832	1,979,999.984953703703703704	0.0000%
3	SushiSwap: Bento Box V1	1,076,891.658251440521472765	0.0000%
4	0xb9fecdd767343ce99a689c5f673839c8174c2e47	989,999.01	0.0000%
5	Null: 0x000...dEaD	54,063.755372484545054481	0.0000%
6	0x9efeefab9106bd314441b6b794b4b9db292d177d	0.014895833333333334	0.0000%

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.

