



SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS REPORT



BASTE

\$BASTE

18/04/2024

TOKEN OVERVIEW

Fees

- Buy fees: 3%
- Sell fees: 3%

Fees privileges

- Can change fees up to 25%

Ownership

- Owned

Minting

- No mint function

Max Tx Amount / Max Wallet Amount

- Can't change max tx amount and max wallet amount

Blacklist

- Blacklist function not 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

BASTE 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 **BASTE** (Customer) to conduct a Smart Contract Code Review and Security Analysis.

0xe29A043A85e82f7548B0CEff341fe88bDd3159a0

Network: **Base Network (ETH)**

This report presents the findings of the security assessment of Customer's smart contract and its code review conducted on **18/04/2024**



WEBSITE DIAGNOSTIC

<https://baste.top/>



0-49



50-89



90-100



Performance



Accessibility



Best
Practices



SEO



Progressive
Web App

Socials



Twitter

<https://twitter.com/BasteCoin>



Telegram

<https://t.me/BasteOnBase>

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't exclude an address from transactions
- Contract owner can exclude/include wallet from tax

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

- Contract owner can exclude/include wallet from rewards

```
function excludeFromReward(address account) public onlyOwner {
    // require(account != 0x7a250d5630B4cF539739dF2C5dAcb4c659F2488D, "We can not exclude Uniswap router.");
    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 setSwapBackSettings(uint256 _amount) external onlyOwner {
    require(
        _amount >= totalSupply().mul(5).div(10**4),
        "Swapback amount should be at least 0.05% of total supply"
    );
    numTokensSellToAddToLiquidity = _amount;
    emit SwapAndLiquifyAmountUpdated(_amount);
}
```

● Contract owner can change fees up to 25%

```
uint256 public constant MAX_FEE = 10**4 / 4;
```

```
function setTaxFeePercent(uint256 taxFeeBps) external onlyOwner {
    _taxFee = taxFeeBps;
    require(
        _taxFee + _liquidityFee + _charityFee <= MAX_FEE,
        "Total fee is over 25%"
    );
}

function setLiquidityFeePercent(uint256 liquidityFeeBps)
    external
    onlyOwner
{
    _liquidityFee = liquidityFeeBps;
    require(
        _taxFee + _liquidityFee + _charityFee <= MAX_FEE,
        "Total fee is over 25%"
    );
}

function setCharityFeePercent(uint256 charityFeeBps) external onlyOwner {
    _charityFee = charityFeeBps;
    require(
        _taxFee + _liquidityFee + _charityFee <= MAX_FEE,
        "Total fee is over 25%"
    );
}
```

● Contract owner can transfer ownership

```
function transferOwnership(address newOwner) public virtual onlyOwner {
    require(newOwner != address(0), "Ownable: new owner is the zero address");
    _setOwner(newOwner);
}

function _setOwner(address newOwner) private {
    address oldOwner = _owner;
    _owner = newOwner;
    emit OwnershipTransferred(oldOwner, newOwner);
}
```

● Contract owner can renounce ownership

```
function renounceOwnership() public virtual onlyOwner {
    _setOwner(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 no HIGH issues 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:	66% unlocked tokens check project Tokenomics section - https://baste.top/



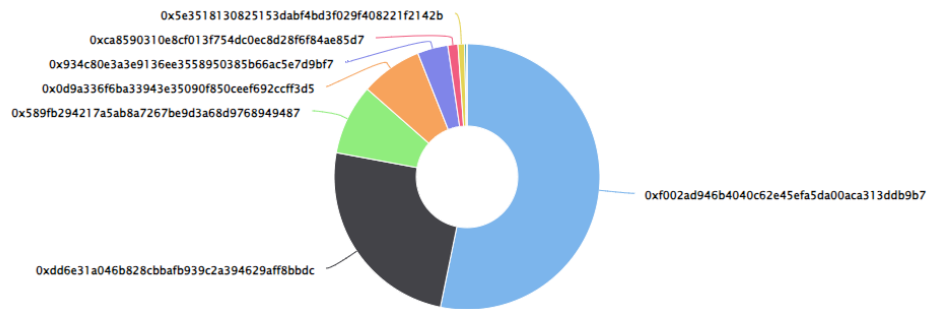
BASTE TOKEN ANALYTICS & TOP 10 TOKEN HOLDERS

The top 10 holders collectively own 100.00% (8,100,000,000.00 Tokens) of BASTE

Token Total Supply: 8,100,000,000.00 Token | Total Token Holders: 8

BASTE Top 10 Token Holders

Source: <https://basescan.org>



(A total of 8,100,000,000.00 tokens held by the top 10 accounts from the total supply of 8,100,000,000.00 token)

Rank	Address	Quantity (Token)	Percentage
1	0xf002ad946b4040c62e45efa5da00aca313ddb9b7	4,310,937,500	53.2215%
2	0xdd6e31a046b828cbbafb939c2a394629aff8bbdc	2,000,000,000	24.6914%
3	0x589fb294217a5ab8a7267be9d3a68d9768949487	693,562,500	8.5625%
4	0xd9a336f6ba33943e35090f850ceef692ccff3d5	607,500,000	7.5000%
5	0x934c80e3a3e9136ee3558950385b66ac5e7d9bf7	300,000,000	3.7037%
6	0xca8590310e8cf013f754dc0ec8d28f6f84ae85d7	100,000,000	1.2346%
7	0x5e3518130825153dabf4bd3f029f408221f2142b	65,000,000	0.8025%
8	0xf1954a30c78e1e5c9afd5abdb893db5b61542a2f	23,000,000	0.2840%

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.

