



Cyberscope

Audit Report

Wall E

April 2022

Type BEP20

Network BSC

Address 0x0799b7ddc8fd8eb92682bd5f366771f0f575fde5

Audited by © cyberscope

Table of Contents

| | |
|---|-----------|
| Table of Contents | 1 |
| Contract Review | 3 |
| Source Files | 3 |
| Audit Updates | 3 |
| Contract Analysis | 4 |
| ST - Stop Transactions | 5 |
| Description | 5 |
| Recommendation | 5 |
| OCTD - Owner Contract Tokens Drain | 7 |
| Description | 7 |
| Recommendation | 7 |
| ELFM - Exceed Limit Fees Manipulation | 8 |
| Description | 8 |
| Recommendation | 8 |
| BC - Blacklisted Contracts | 9 |
| Description | 9 |
| Recommendation | 9 |
| Contract Diagnostics | 10 |
| L01 - Public Function could be Declared External | 11 |
| Description | 11 |
| Recommendation | 11 |
| L02 - State Variables could be Declared Constant | 12 |
| Description | 12 |
| Recommendation | 12 |
| L04 - Conformance to Solidity Naming Conventions | 13 |
| Description | 13 |

| | |
|---|-----------|
| Recommendation | 13 |
| L07 - Missing Events Arithmetic | 14 |
| Description | 14 |
| Recommendation | 14 |
| L09 - Dead Code Elimination | 15 |
| Description | 15 |
| Recommendation | 15 |
| L13 - Divide before Multiply Operation | 16 |
| Description | 16 |
| Recommendation | 16 |
| Contract Functions | 17 |
| Contract Flow | 20 |
| Domain Info | 21 |
| Summary | 22 |
| Disclaimer | 23 |
| About Cyberscope | 24 |

Contract Review

| | |
|-------------------------|---|
| Contract Name | CMW |
| Compiler Version | v0.8.13+commit.abaa5c0e |
| Optimization | 200 runs |
| Licence | None |
| Explorer | https://bscscan.com/token/0x0799b7DDC8fD8eB92682bD5F366771F0f575FdE5 |
| Symbol | WE |
| Decimals | 9 |
| Total Supply | 10,000,000,000,000 |
| Domain | wall-e.online |

Source Files

| | |
|---------------------|--|
| Filename | SHA256 |
| contract.sol | 791b740d67732f1ddf0df5ef300914292c72f7ca44ebdfdf9a56c206bb3fff1c |

Audit Updates

| | |
|----------------------|-----------------|
| Initial Audit | 17th April 2022 |
| Corrected | |

Contract Analysis

● Critical ● Medium ● Minor ● Pass

| Severity | Code | Description |
|----------|------|---|
| ● | ST | Contract Owner is not able to stop or pause transactions |
| ● | OCTD | Contract Owner is not able to transfer tokens from specific address |
| ● | OTUT | Owner Transfer User's Tokens |
| ● | ELFM | Contract Owner is not able to increase fees more than a reasonable percent (25%) |
| ● | ULTW | Contract Owner is not able to increase the amount of liquidity taken by dev wallet more than a reasonable percent |
| ● | MT | Contract Owner is not able to mint new tokens |
| ● | BT | Contract Owner is not able to burn tokens from specific wallet |
| ● | BC | Contract Owner is not able to blacklist wallets from selling |

ST - Stop Transactions

| | |
|-------------|---------------------------|
| Criticality | critical |
| Location | contract.sol#L292,294,455 |

Description

The contract owner has the authority to stop transactions for all users excluding the owner. The owner may take advantage of it by setting the `_maxTxAmount` to zero.

```
function checkTxLimit(address sender, uint256 amount) internal view {
    require(amount <= _maxTxAmount || isTxLimitExempt[sender], "TX Limit Exceeded");
}
```

The contract owner has the authority to stop transactions for all users excluding the owner. The owner may take advantage of it by setting the `tradingOpen` to false.

```
if(!_intAddr[sender] && !_intAddr[recipient]){
    require(tradingOpen,"Trading not open yet");
}
```

The contract owner has the authority to convert the contract into a **HONEYPOT** and prevent users from selling. The owner may take advantage of it by setting the `_sellMultiplier` to high value.

```
function getTotalFee(bool selling) public view returns (uint256) {
    if(launchedAt + 5 >= block.number){ return feeDenominator.sub(1); }
    if(selling) { return totalFee.mul(_sellMultiplier); }
    return totalFee;
}
```

Recommendation

The contract could embody a check for not allowing setting the `_maxTxAmount` and the `_sellMultiplier` less than a reasonable amount. A suggested implementation could check that the maximum amount should be more than a fixed percentage of the total supply.

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. That risk can be prevented by temporarily locking the contract or renouncing ownership.

OCTD - Owner Contract Tokens Drain

| | |
|--------------------|-------------------|
| Criticality | minor |
| Location | contract.sol#L475 |

Description

The contract owner has the authority to claim all the balance of the contract. The owner may take advantage of it by calling the `manualSend` function.

```
function manualSend() external authorized {  
    uint256 contractETHBalance = address(this).balance;  
    payable(marketingFeeReceiver).transfer(contractETHBalance);  
}
```

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. That risk can be prevented by temporarily locking the contract or renouncing ownership.

ELFM - Exceed Limit Fees Manipulation

| | |
|-------------|-------------------|
| Criticality | minor |
| Location | contract.sol#L433 |

Description

The contract owner has the authority to increase over the allowed limit of 25%. The owner may take advantage of it by calling the `setFees` function and increase them up to a total 33%.

```
function setFees(uint256 _liquidityFee, uint256 _marketingFee, uint256
_devFee, uint256 _feeDenominator) external authorized {
    liquidityFee = _liquidityFee;
    marketingFee = _marketingFee;
    devFee = _devFee;
    totalFee = _liquidityFee.add(_marketingFee).add(_devFee);
    feeDenominator = _feeDenominator;
    require(totalFee < feeDenominator/3);
}
```

Recommendation

The contract could embody a check for the maximum acceptable value.

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. That risk can be prevented by temporarily locking the contract or renouncing ownership.

BC - Blacklisted Contracts

| | |
|--------------------|-------------------|
| Criticality | medium |
| Location | contract.sol#L296 |

Description

The contract owner has the authority to stop contracts from transactions. The owner may take advantage of it by calling the `blacklistAddress` function.

```
require(!isBlacklisted[recipient] && !isBlacklisted[sender], 'Address  
is blacklisted');
```

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. That risk can be prevented by temporarily locking the contract or renouncing ownership.

Contract Diagnostics

● Critical ● Medium ● Minor

| Severity | Code | Description |
|----------|------|--|
| ● | L01 | Public Function could be Declared External |
| ● | L02 | State Variables could be Declared Constant |
| ● | L04 | Conformance to Solidity Naming Conventions |
| ● | L07 | Missing Events Arithmetic |
| ● | L09 | Dead Code Elimination |
| ● | L13 | Divide before Multiply Operation |

L01 - Public Function could be Declared External

| | |
|--------------------|--|
| Criticality | minor |
| Location | contract.sol#L95,103,124,350,442,472,480,494 |

Description

Public functions that are never called by the contract should be declared external to save gas.

```
isOverLiquified  
transferForeignToken  
blacklistAddress  
cooldownEnabled  
tradingStatus  
transferOwnership  
unauthorize  
authorize
```

Recommendation

Use the external attribute for functions never called from the contract.

L02 - State Variables could be Declared Constant

| | |
|--------------------|-----------------------------------|
| Criticality | minor |
| Location | contract.sol#L189,188,190,197,191 |

Description

Constant state variables should be declared constant to save gas.

```
routerAddress
_totalSupply
ZERO
WBNB
DEAD
```

Recommendation

Add the constant attribute to state variables that never change.

L04 - Conformance to Solidity Naming Conventions

Criticality

minor

Location

contract.sol#L71,139,350,433,442,458,467,472,480,188,189,190,193,194,195,197,198,199,201,202,214

Description

Solidity defines a naming convention that should be followed. Rule exceptions:

- Allow constant variable name/symbol/decimals to be lowercase.
- Allow _ at the beginning of the mixed_case match for private variables and unused parameters.

```
_sellMultiplier  
_allowances  
_balances  
_maxWalletSize  
_maxTxAmount  
_totalSupply  
_decimals  
_symbol  
_name  
...
```

Recommendation

Follow the Solidity naming convention.

<https://docs.soliditylang.org/en/v0.4.25/style-guide.html#naming-conventions>

L07 - Missing Events Arithmetic

Criticality

minor

Location

contract.sol#L433,455,463

Description

Detected missing events for critical arithmetic parameters. There are functions that have no event emitted, so it is difficult to track off-chain changes.

```
_maxTxAmount = amountBuy  
_sellMultiplier = multiplier  
liquidityFee = _liquidityFee
```

Recommendation

Emit an event for critical parameter changes.

L09 - Dead Code Elimination

| | |
|--------------------|-------------------|
| Criticality | minor |
| Location | contract.sol#L406 |

Description

Functions that are not used in the contract, and make the code's size bigger.

```
buyTokens
```

Recommendation

Remove unused functions.

L13 - Divide before Multiply Operation

| | |
|--------------------|-------------------|
| Criticality | minor |
| Location | contract.sol#L185 |

Description

Performing divisions before multiplications may cause lose of prediction.

```
swapThreshold = _totalSupply / 1000 * 1
```

Recommendation

The multiplications should be prior to the divisions.

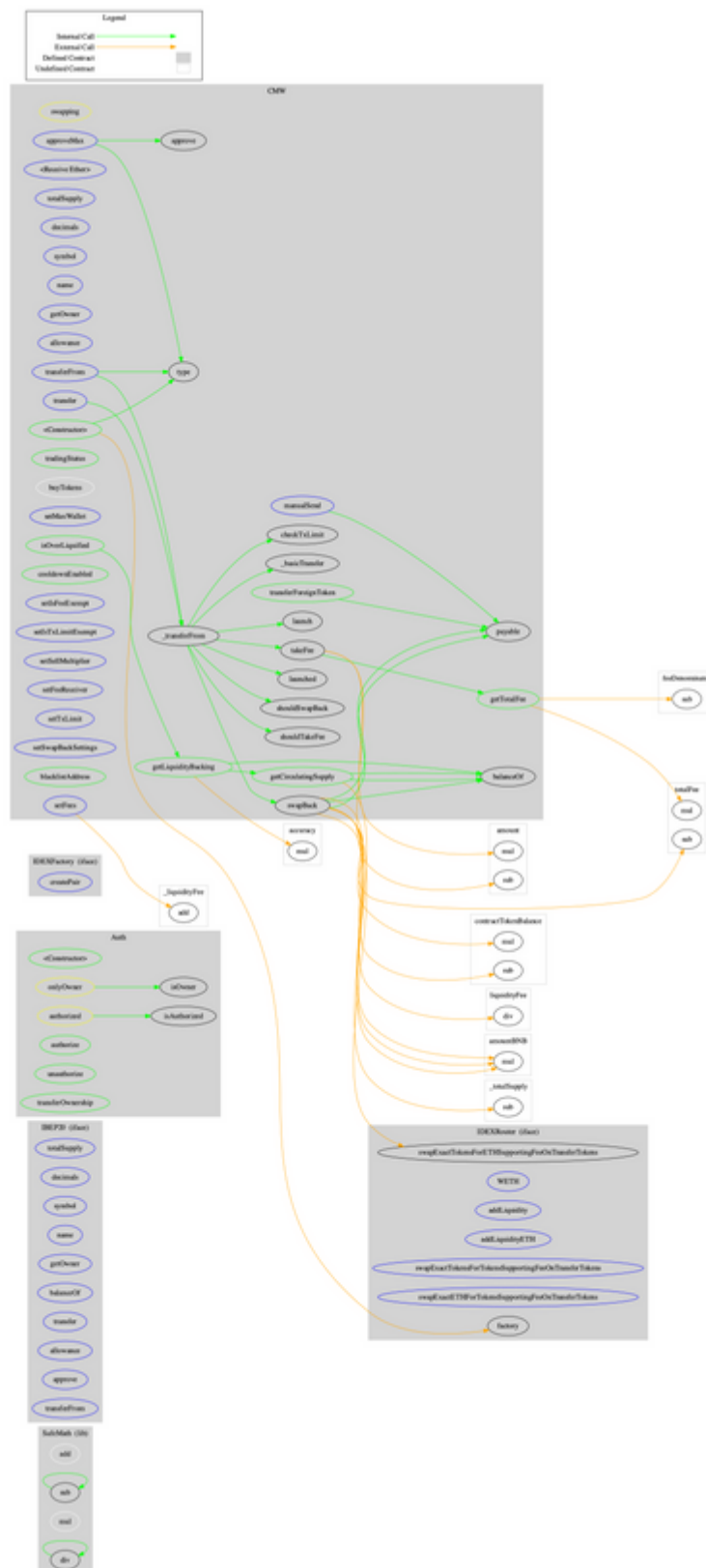
Contract Functions

| Contract | Type | Bases | | |
|--------------------|-------------------|------------|------------|-----------|
| | Function Name | Visibility | Mutability | Modifiers |
| | | | | |
| SafeMath | Library | | | |
| | add | Internal | | |
| | sub | Internal | | |
| | sub | Internal | | |
| | mul | Internal | | |
| | div | Internal | | |
| | div | Internal | | |
| | | | | |
| IBEP20 | Interface | | | |
| | totalSupply | External | | - |
| | decimals | External | | - |
| | symbol | External | | - |
| | name | External | | - |
| | getOwner | External | | - |
| | balanceOf | External | | - |
| | transfer | External | ✓ | - |
| | allowance | External | | - |
| | approve | External | ✓ | - |
| | transferFrom | External | ✓ | - |
| | | | | |
| Auth | Implementation | | | |
| | <Constructor> | Public | ✓ | - |
| | authorize | Public | ✓ | onlyOwner |
| | unauthorize | Public | ✓ | onlyOwner |
| | isOwner | Public | | - |
| | isAuthorized | Internal | | |
| | transferOwnership | Public | ✓ | onlyOwner |
| | | | | |
| IDEXFactory | Interface | | | |

| | | | | |
|-------------------|---|--------------|---------|------------|
| | createPair | External | ✓ | - |
| | | | | |
| IDEXRouter | Interface | | | |
| | factory | External | | - |
| | WETH | External | | - |
| | addLiquidity | External | ✓ | - |
| | addLiquidityETH | External | Payable | - |
| | swapExactTokensForTokensSupportingFeeOnTransferTokens | External | ✓ | - |
| | swapExactETHForTokensSupportingFeeOnTransferTokens | External | Payable | - |
| | swapExactTokensForETHSupportingFeeOnTransferTokens | External | ✓ | - |
| | | | | |
| CMW | Implementation | IBEP20, Auth | | |
| | <Constructor> | Public | ✓ | Auth |
| | <Receive Ether> | External | Payable | - |
| | totalSupply | External | | - |
| | decimals | External | | - |
| | symbol | External | | - |
| | name | External | | - |
| | getOwner | External | | - |
| | balanceOf | Public | | - |
| | allowance | External | | - |
| | approve | Public | ✓ | - |
| | approveMax | External | ✓ | - |
| | transfer | External | ✓ | - |
| | transferFrom | External | ✓ | - |
| | _transferFrom | Internal | ✓ | |
| | _basicTransfer | Internal | ✓ | |
| | checkTxLimit | Internal | | |
| | shouldTakeFee | Internal | | |
| | getTotalFee | Public | | - |
| | takeFee | Internal | ✓ | |
| | tradingStatus | Public | ✓ | authorized |
| | shouldSwapBack | Internal | | |

| | | | | |
|--|----------------------|----------|---|------------|
| | swapBack | Internal | ✓ | swapping |
| | buyTokens | Internal | ✓ | swapping |
| | launched | Internal | | |
| | launch | Internal | ✓ | |
| | setMaxWallet | External | ✓ | onlyOwner |
| | setFees | External | ✓ | authorized |
| | cooldownEnabled | Public | ✓ | authorized |
| | setIsFeeExempt | External | ✓ | authorized |
| | setIsTxLimitExempt | External | ✓ | authorized |
| | setSellMultiplier | External | ✓ | authorized |
| | setFeeReceiver | External | ✓ | authorized |
| | setTxLimit | External | ✓ | authorized |
| | setSwapBackSettings | External | ✓ | authorized |
| | blacklistAddress | Public | ✓ | authorized |
| | manualSend | External | ✓ | authorized |
| | transferForeignToken | Public | ✓ | authorized |
| | getCirculatingSupply | Public | | - |
| | getLiquidityBacking | Public | | - |
| | isOverLiquified | Public | | - |

Contract Flow



Domain Info

| | |
|-------------------------------|---|
| Domain Name | wall-e.online |
| Registry Domain ID | D290306079-CNIC |
| Creation Date | 2022-04-12T06:47:54+00:00 |
| Updated Date | 2022-04-12T06:47:55+00:00 |
| Registry Expiry Date | 2023-04-12T23:59:59+00:00 |
| Registrar WHOIS Server | whois.godaddy.com |
| Registrar URL | https://www.godaddy.com/ |
| Registrar | Go Daddy, LLC |
| Registrar IANA ID | 146 |

The domain has been created 5 days before the creation of the audit. It will expire in 12 months.

There is no public billing information, the creator is protected by the privacy settings.

Summary

Wall E Token is an interesting project that has a friendly and growing community. There are some functions that can be abused by the owner, like manipulating fees and transferring funds to the team's wallet. The owner can blacklist wallets from trading, and increase selling fees up to 100% converting the contract into a **HONEYPOT**. A multi-wallet signing pattern will provide security against potential hacks. Temporarily locking the contract or renouncing ownership will eliminate all the contract threats.

Disclaimer

All the content provided in this document is for general information only and should not be used as financial advice or a reason to buy any investment.

Cyberscope team provides no guarantees against the sale of team tokens or the removal of liquidity by the project audited in this document. Always Do your own research and protect yourselves from being scammed.

The Cyberscope team has audited this project for general information and only expresses their opinion based on similar projects and checks from popular diagnostic tools. Under no circumstances did Cyberscope receive a payment to manipulate those results or change the awarding badge that we will be adding in our website.

Always Do your own research and protect yourselves from scams. This document should not be presented as a reason to buy or not buy any particular token.

The Cyberscope team disclaims any liability for the resulting losses.

About Cyberscope

Coinscope audit and K.Y.C. service has been rebranded to Cyberscope.

Cyberscope is the leading early coin listing, voting and auditing authority firm. The audit process is analyzing and monitoring many aspects of the project. That way, it gives the community a good sense of security using an informative report and a generic score.

Cyberscope and Coinscope are aiming to make crypto discoverable and efficient globally. They provides all the essential tools to assist users draw their own conclusions.



The Cyberscope team

<https://www.cyberscope.io>