
# Audit Report

# **Glory Allstar**

February 2022

| | |
|---|---|
| Type | BEP20 |
| Network | BSC |
| Address | 0x8d721B0Fd5075005fc22BC7d4983d5Bb8a2e761D |
| Audited by | © coinscope |

# Table of Contents

# Contract Review

| | |
|---|---|
| **Contract Name** | MoBaToken |
| **Compiler Version** | v0.8.7+commit.e28d00a7 |
| **Optimization** | 200 runs |
| **Licence** | MIT |
| **Explorer** | https://bscscan.com/token/0x8d721B0Fd5075005fc22BC7d4983d5Bb8a2e761D |
| **Symbol** | MoBa |
| **Decimals** | 18 |
| **Total Supply** | - |
| **Source** | contract.sol |
| **Domain** | gloryallstar.io |

# Audit Updates

| | |
|---|---|
| **Initial Audit** | 16th February 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 |

# MT - Mint Tokens

| | |
|---|---|
| **Criticality** | critical |
| **Location** | contract.sol#L799 |

## Description

The contract owner has the authority to mint tokens. The owner may take advantage of it by calling the `cross` function with a small amount, and after some period call it again with a huge amount. As a result the contract tokens will be highly inflated.

```
function cross(address account, uint256 amount) onlyOwner public {
    require(totalSupply().add(amount) <= cap, "exceeds.");
    _mint(account, amount);
}
```

## Recommendation

The owner should carefully manage the credentials of the owner's account. We advised considering an extra-strong security mechanism that the actions may be quarantined by many users instead of one. The owner could also renounce the contract ownership for a period of time or pass the access to the zero address.

# BT - Burn Tokens

| | |
|---|---|
| **Criticality** | critical |
| **Location** | contract.sol#L804 |

## Description

The contract owner has the authority to burn tokens from a specific address. The owner may take advantage of it by calling the `managerBurn` function. As a result the targeted contract address will lose the corresponding tokens.

```
function managerBurn(address account, uint256 amount) onlyOwner public {
    require(totalSupply().sub(amount) >= 0, "exceeds.");
    _burn(account, amount);
}
```

## Recommendation

The owner should carefully manage the credentials of the owner's account. We advised considering an extra-strong security mechanism that the actions may be quarantined by many users instead of one. The owner could also renounce the contract ownership for a period of time or pass the access to the zero address.

# Contract Diagnostics

Critical Medium Minor

| Severity | Code | Description |
|---|---|---|
| | CO | Code Optimization |
| | L01 | Public Function could be Declared External |
| | L02 | State Variables could be Declared Constant |
| | L05 | Unused State Variable |
| | L04 | Conformance to Solidity Naming Conventions |
| | L09 | Dead Code Elimination |
| | L12 | Using Variables before Declaration |
| | L15 | Local Scope Variable Shadowing |
| | L14 | Uninitialized Variables in Local Scope |

# CO - Code Optimization

| | |
|---|---|
| **Criticality** | minor |
| **Location** | contract.sol#L1595 |

## Description

There are code segments that could be optimized. A segment may be optimized so that it becomes a smaller size, consumes less memory, executes more rapidly, or performs fewer operations.

The “swapTokensForEth()” is called four times during the swap & liquify process. The repetitive call produced a lot of gas fees.

```
uint256 marketingTokens = contractTokenBalance
.mul(marketingFee)
.div(totalFees);
swapAndSendDividendsToMarketing(marketingTokens);

uint256 buyBackTokens = contractTokenBalance
.mul(buyBackFee)
.div(totalFees);
swapAndSendDividendsToBuyBackAddress(buyBackTokens);

uint256 swapTokens = contractTokenBalance.mul(liquidityFee).div(
    totalFees
);
swapAndLiquify(swapTokens);

uint256 sellTokens = balanceOf(address(this));
swapAndSendDividends(sellTokens);
```

## Recommendation

There could be one call to the “swapTokensForEth()” function that swaps the accumulated amount. Then the total amount could be splitted proportionally.

# L01 - Public Function could be Declared External

| | |
|---|---|
| **Criticality** | minor |
| **Location** | contract.sol#L284,293,676,684,701,727,735,746,764,782 and 18 more |

## Description

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

```
process
getAccountAtIndex
dividendTokenBalanceOf
withdrawableDividendOf
isExcludedFromFees
updateGasForProcessing
setAutomatedMarketMakerPair
excludeMultipleAccountsFromFees
updateUniswapV2Router
...
```

## Recommendation

Use the external attribute for functions never called from the contract

# L02 - State Variables could be Declared Constant

| | |
|---|---|
| **Criticality** | minor |
| **Location** | contract.sol#L2000,1999,1997,1988 |

## Description

Constant state variables should be declared constant to save gas.

```
_tokenSupply
_routerAddress
_marketingAddress
_buyBackAddress
```

## Recommendation

Add the constant attribute to state variables that never change.

# L05 - Unused State Variable

| | |
|---|---|
| **Criticality** | minor |
| **Location** | contract.sol#L306 |

## Description

There are segments that contain unused state variables.

```
MAX_INT256
```

## Recommendation

Remove unused state variables.

# L04 - Conformance to Solidity Naming Conventions

| | |
|---|---|
| **Criticality** | minor |
| **Location** | contract.sol#L18,175,176,193,1138,1145,1152,1162,1053,1818 |

## 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.

```
_account
magnitude
_owner
MINIMUM_LIQUIDITY
PERMIT_TYPEHASH
DOMAIN_SEPARATOR
WETH
...
```

## Recommendation

Follow the Solidity naming convention.
https://docs.soliditylang.org/en/v0.4.25/style-guide.html#naming-conventions

# L09 - Dead Code Elimination

| | |
|---|---|
| **Criticality** | minor |
| **Location** | contract.sol#L242,1172,899,548,510,568,382,418,428,403 and 4 more |

## Description

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

```
mul
div
abs
trySub
tryMul
tryMod
tryDiv
tryAdd
mod
...
```

## Recommendation

Remove unused functions.

# L12 - Using Variables before Declaration

| | |
|---|---|
| **Criticality** | minor |
| **Location** | contract.sol#L1641,1640,1642 |

## Description

The contract is using a variable before the declaration. This is usually happening either if it has not been declared yet or the variable has been declared in a different scope.

```
lastProcessedIndex
iterations
claims
```

## Recommendation

The variables should be declared before any usage of them.

# L15 - Local Scope Variable Shadowing

| | |
|---|---|
| **Criticality** | minor |
| **Location** | contract.sol#L1073 |

## Description

The are variables that are defined in the local scope containing the same name from an upper scope.

```
_symbol
_name
```

## Recommendation

The local variables should have different names from the upper scoped variables.

# L14 - Uninitialized Variables in Local Scope

| | |
|---|---|
| **Criticality** | minor |
| **Location** | contract.sol#L1640,1641,1642 |

## Description

The are variables that are defined in the local scope and are not initialized.

```
lastProcessedIndex
claims
iterations
```

## Recommendation

All the local scoped variables should be initialized.

# Contract Functions

| Contract | Type | Bases | | |
|---|---|---|---|---|
| | **Function Name** | **Visibility** | **Mutability** | **Modifiers** |
| | | | | |
| **IUniswapV2Router01** | Interface | | | |
| | factory | External | | - |
| | WETH | External | | - |
| | addLiquidity | External | ✓ | - |
| | addLiquidityETH | External | Payable | - |
| | removeLiquidity | External | ✓ | - |
| | removeLiquidityETH | External | ✓ | - |
| | removeLiquidityWithPermit | External | ✓ | - |
| | removeLiquidityETHWithPermit | External | ✓ | - |
| | swapExactTokensForTokens | External | ✓ | - |
| | swapTokensForExactTokens | External | ✓ | - |
| | swapExactETHForTokens | External | Payable | - |
| | swapTokensForExactETH | External | ✓ | - |
| | swapExactTokensForETH | External | ✓ | - |
| | swapETHForExactTokens | External | Payable | - |
| | quote | External | | - |
| | getAmountOut | External | | - |
| | getAmountIn | External | | - |
| | getAmountsOut | External | | - |
| | getAmountsIn | External | | - |
| | | | | |
| **IUniswapV2Router02** | Interface | IUniswapV2Router01 | | |
| | removeLiquidityETHSupportingFeeOnTransferTokens | External | ✓ | - |
| | removeLiquidityETHWithPermitSupportingFeeOnTransferTokens | External | ✓ | - |
| | swapExactTokensForTokensSupportingFeeOnTransferTokens | External | ✓ | - |

| | | | | |
|---|---|---|---|---|
| | swapExactETHForTokensSupportingFeeOnTransferTokens | External | Payable | - |
| | swapExactTokensForETHSupportingFeeOnTransferTokens | External | ✓ | - |
| | | | | |
| **IUniswapV2Pair** | Interface | | | |
| | name | External | | - |
| | symbol | External | | - |
| | decimals | External | | - |
| | totalSupply | External | | - |
| | balanceOf | External | | - |
| | allowance | External | | - |
| | approve | External | ✓ | - |
| | transfer | External | ✓ | - |
| | transferFrom | External | ✓ | - |
| | DOMAIN_SEPARATOR | External | | - |
| | PERMIT_TYPEHASH | External | | - |
| | nonces | External | | - |
| | permit | External | ✓ | - |
| | MINIMUM_LIQUIDITY | External | | - |
| | factory | External | | - |
| | token0 | External | | - |
| | token1 | External | | - |
| | getReserves | External | | - |
| | price0CumulativeLast | External | | - |
| | price1CumulativeLast | External | | - |
| | kLast | External | | - |
| | mint | External | ✓ | - |
| | burn | External | ✓ | - |
| | swap | External | ✓ | - |
| | skim | External | ✓ | - |
| | sync | External | ✓ | - |
| | initialize | External | ✓ | - |
| | | | | |
| **IUniswapV2Factory** | Interface | | | |

| | | | | |
|---|---|---|---|---|
| | feeTo | External | | - |
| | feeToSetter | External | | - |
| | getPair | External | | - |
| | allPairs | External | | - |
| | allPairsLength | External | | - |
| | createPair | External | ✓ | - |
| | setFeeTo | External | ✓ | - |
| | setFeeToSetter | External | ✓ | - |
| | | | | |
| **Context** | Implementation | | | |
| | _msgSender | Internal | | |
| | _msgData | Internal | | |
| | | | | |
| **Ownable** | Implementation | Context | | |
| | <Constructor> | Public | ✓ | - |
| | owner | Public | | - |
| | renounceOwnership | Public | ✓ | onlyOwner |
| | transferOwnership | Public | ✓ | onlyOwner |
| | | | | |
| **SafeMathInt** | Library | | | |
| | mul | Internal | | |
| | div | Internal | | |
| | sub | Internal | | |
| | add | Internal | | |
| | abs | Internal | | |
| | toUint256Safe | Internal | | |
| | | | | |
| **SafeMathUint** | Library | | | |
| | toInt256Safe | Internal | | |
| | | | | |
| **SafeMath** | Library | | | |
| | tryAdd | Internal | | |
| | trySub | Internal | | |
| | tryMul | Internal | | |
| | tryDiv | Internal | | |

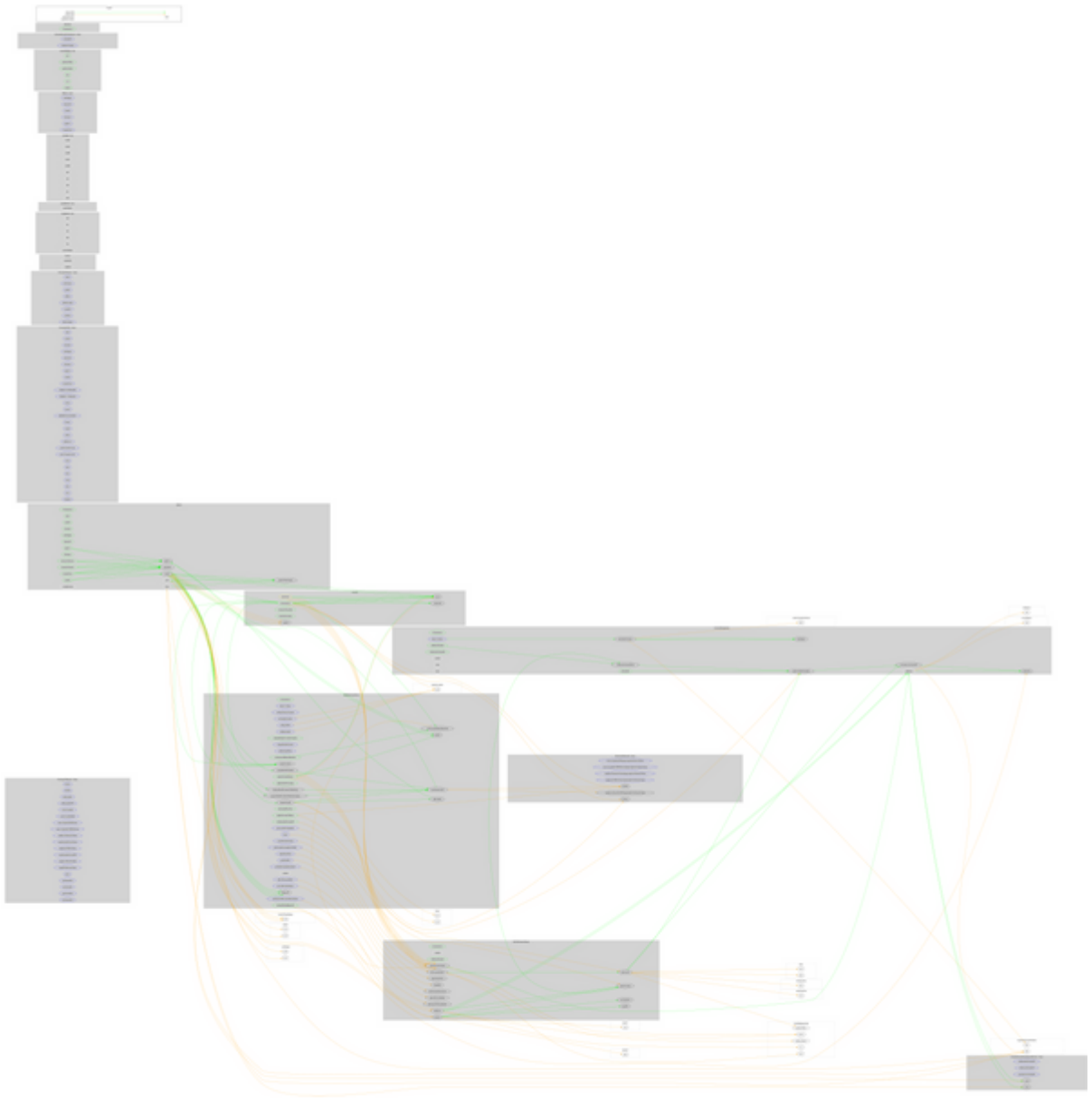
| | | | | |
|---|---|---|---|---|
| | tryMod | Internal | | |
| | add | Internal | | |
| | sub | Internal | | |
| | mul | Internal | | |
| | div | Internal | | |
| | mod | Internal | | |
| | sub | Internal | | |
| | div | Internal | | |
| | mod | Internal | | |
| | | | | |
| **IERC20** | Interface | | | |
| | totalSupply | External | | - |
| | balanceOf | External | | - |
| | transfer | External | ✓ | - |
| | allowance | External | | - |
| | approve | External | ✓ | - |
| | transferFrom | External | ✓ | - |
| | | | | |
| **ERC20** | Implementation | Context, IERC20 | | |
| | <Constructor> | Public | ✓ | - |
| | name | Public | | - |
| | symbol | Public | | - |
| | decimals | Public | | - |
| | totalSupply | Public | | - |
| | balanceOf | Public | | - |
| | transfer | Public | ✓ | - |
| | allowance | Public | | - |
| | approve | Public | ✓ | - |
| | transferFrom | Public | ✓ | - |
| | increaseAllowance | Public | ✓ | - |
| | decreaseAllowance | Public | ✓ | - |
| | _transfer | Internal | ✓ | |
| | _mint | Internal | ✓ | |
| | _burn | Internal | ✓ | |
| | _approve | Internal | ✓ | |

| | | | | |
|---|---|---|---|---|
| | _setupDecimals | Internal | ✓ | |
| | _beforeTokenTransfer | Internal | ✓ | |
| | | | | |
| **IterableMapping** | Library | | | |
| | get | Public | | - |
| | getIndexOfKey | Public | | - |
| | getKeyAtIndex | Public | | - |
| | size | Public | | - |
| | set | Public | ✓ | - |
| | remove | Public | ✓ | - |
| | | | | |
| **DividendPayingTokenInterface** | Interface | | | |
| | dividendOf | External | | - |
| | withdrawDividend | External | ✓ | - |
| | | | | |
| **DividendPayingTokenOptionalInterface** | Interface | | | |
| | withdrawableDividendOf | External | | - |
| | withdrawnDividendOf | External | | - |
| | accumulativeDividendOf | External | | - |
| | | | | |
| **DividendPayingToken** | Implementation | ERC20, DividendPayingTokenInterface, DividendPayingTokenOptionalInterface | | |
| | <Constructor> | Public | ✓ | ERC20 |
| | <Receive Ether> | External | Payable | - |
| | distributeDividends | Public | Payable | - |
| | withdrawDividend | Public | ✓ | - |
| | _withdrawDividendOfUser | Internal | ✓ | |
| | dividendOf | Public | | - |

| | | | | |
|---|---|---|---|---|
| | withdrawableDividendOf | Public | | - |
| | withdrawnDividendOf | Public | | - |
| | accumulativeDividendOf | Public | | - |
| | _transfer | Internal | ✓ | |
| | _mint | Internal | ✓ | |
| | _burn | Internal | ✓ | |
| | _setBalance | Internal | ✓ | |
| | | | | |
| **ERC20DividendToken** | Implementation | ERC20, Ownable | | |
| | <Constructor> | Public | ✓ | ERC20 |
| | <Receive Ether> | External | Payable | - |
| | setSwapTokensAtAmount | External | ✓ | onlyOwner |
| | updateDividendTracker | Public | ✓ | onlyOwner |
| | updateUniswapV2Router | Public | ✓ | onlyOwner |
| | excludeFromFees | Public | ✓ | onlyOwner |
| | excludeMultipleAccountsFromFees | Public | ✓ | onlyOwner |
| | changeMaxSellAmount | External | ✓ | onlyOwner |
| | setMarketingWallet | External | ✓ | onlyOwner |
| | setTokenRewardsFee | External | ✓ | onlyOwner |
| | setLiquiditFee | External | ✓ | onlyOwner |
| | setMarketingFee | External | ✓ | onlyOwner |
| | setAutomatedMarketMakerPair | Public | ✓ | onlyOwner |
| | _setAutomatedMarketMakerPair | Private | ✓ | |
| | updateGasForProcessing | Public | ✓ | onlyOwner |
| | updateClaimWait | External | ✓ | onlyOwner |
| | getClaimWait | External | | - |
| | getTotalDividendsDistributed | External | | - |
| | isExcludedFromFees | Public | | - |
| | withdrawableDividendOf | Public | | - |
| | dividendTokenBalanceOf | Public | | - |
| | excludeFromDividends | External | ✓ | onlyOwner |
| | getAccountDividendsInfo | External | | - |
| | getAccountDividendsInfoAtIndex | External | | - |
| | processDividendTracker | External | ✓ | - |
| | claim | External | ✓ | - |

| | | | | |
|---|---|---|---|---|
| | getLastProcessedIndex | External | | - |
| | getNumberOfDividendTokenHolders | External | | - |
| | _transfer | Internal | ✓ | |
| | swapAndLiquify | Private | ✓ | |
| | swapTokensForEth | Private | ✓ | |
| | addLiquidity | Private | ✓ | |
| | swapAndSendDividends | Private | ✓ | |
| | swapAndSendDividendsToMarketing | Private | ✓ | |
| | swapAndSendDividendsToBuyBackAddress | Private | ✓ | |
| | | | | |
| **ERC20DividendTracker** | Implementation | Ownable, DividendPayingToken | | |
| | <Constructor> | Public | ✓ | DividendPayingToken |
| | _transfer | Internal | | |
| | withdrawDividend | Public | | - |
| | excludeFromDividends | External | ✓ | onlyOwner |
| | updateClaimWait | External | ✓ | onlyOwner |
| | getLastProcessedIndex | External | | - |
| | getNumberOfTokenHolders | External | | - |
| | getAccount | Public | | - |
| | getAccountAtIndex | Public | | - |
| | canAutoClaim | Private | | |
| | setBalance | External | ✓ | onlyOwner |
| | process | Public | ✓ | - |
| | processAccount | Public | ✓ | onlyOwner |
| | | | | |
| **BabyZilla** | Implementation | ERC20DividendToken | | |
| | <Constructor> | Public | ✓ | ERC20DividendToken |

# Contract Flow



# Domain Info

| | |
|---|---|
| **Domain Name** | gloryallstar.io |
| **Registry Domain ID** | bc33254e0daf402e9ae4a711300ad9d6-DONUTS |
| **Creation Date** | 2021-11-09T13:12:39Z |
| **Updated Date** | 2021-11-23T12:15:52Z |
| **Registry Expiry Date** | 2023-11-09T13:12:39Z |
| **Registrar WHOIS Server** | whois.godaddy.com/ |
| **Registrar URL** | http://www.godaddy.com/domains/search.aspx?ci=8990 |
| **Registrar** | GoDaddy.com, LLC |
| **Registrar IANA ID** | 146 |

The domain has been created 3 months before the creation of the audit. It will expire in over 1 year.

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

# Summary

The contract owner has the authority to mint or burn tokens on a specific address. If this functionality is abused by the contract owner, the contract tokens will be highly inflated or deflated. 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.

Coinscope 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 Coinscope 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 Coinscope 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 Coinscope team disclaims any liability for the resulting losses.

# About Coinscope

Coinscope 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.

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

The Coinscope.co team

https://www.coinscope.co