RUGFREECOINS

BSC: Baby Golden Coin

RugfreeCoins Verified on March 15th, 2024

Overview

✅ No mint function found, the owner cannot mint tokens after initial deployment.

✅ The owner can’t set a max transaction limit

✅ The owner can’t enable or pause trading

✅ The owner can't change fees.

✅ The owner can't blacklist wallets.

✅ The owner can’t set a max wallet limit

✅ The owner can’t claim the contract’s balance of its own token.

Contents

Audit details

Audited project

Baby Golden Coin

Contract Address

0x85cEf71A3277220cc3D88A1703Abc771cAA2241F

Client contact

Baby Golden Coin Team

Blockchain

Binance Smart chain

Project website

https://babygoldencoin.com/

Disclaimer

This is a limited report on our findings based on our analysis, in accordance with good industry practice as at the date of this report, in relation to cybersecurity vulnerabilities and issues in the framework and algorithms based on smart contracts, the details of which are set out in this report. In order to get a full view of our analysis, it is crucial for you to read the full report. While we have done our best in conducting our analysis and producing this report, it is important to note that you should not rely on this report and cannot claim against us on the basis of what it says or doesn't say, or how we produced it, and it is important for you to conduct your own independent investigations before making any decisions. We go into more detail on this in the disclaimer below – **please make sure to read it in full.**

ⓘ DISCLAIMER

By reading this report or any part of it, you agree to the terms of this disclaimer. If you do not agree to the terms, then please immediately cease reading this report, and delete and destroy any and all copies of this report downloaded and/or printed by you. **This report is provided for information purposes only and on a non-reliance basis, and does not constitute investment advice.** No one shall have any right to rely on the report or its contents, and **RugfreeCoins and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (RugfreeCoins) owe no duty of care towards you or any other person**, nor does RugfreeCoins make any warranty or representation to any person on the accuracy or completeness of the report.

The report is provided "as is", without any conditions, warranties or other terms of any kind except as set out in this disclaimer, and RugfreeCoins hereby excludes all representations, warranties, conditions and other terms (including, without limitation, the warranties implied by law of satisfactory quality, fitness for purpose and the use of reasonable care and skill) which, but for this clause, might have effect in relation to the report. Except and only to the extent that it is prohibited by law, RugfreeCoins hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against RugfreeCoins, for any amount or kind of loss or damage that may result to you or any other person (including without limitation, any direct, indirect, special, punitive, consequential or pure economic loss or damages, or any loss of income, profits, goodwill, data, contracts, use of money, or business interruption, and whether in delict, tort (including without limitation negligence), contract, breach of statutory duty, misrepresentation (whether innocent or negligent) or otherwise under any claim of any nature whatsoever in any jurisdiction) in any way arising from or connected with this report and the use, inability to use or the results of use of this report, and any reliance on this report.

The analysis of the security is purely based on the smart contracts alone. No applications or operations were reviewed for security. No product code has been reviewed.

Background

RugfreeCoins was commissioned by the **Baby Golden Coin Team** to perform an audit of the smart contract.

https://bscscan.com/token/0x85cef71a3277220cc3d88a1703abc771caa2241f

This audit focuses on verifying that the smart contract is secure, resilient, and working according to the specifications.

The information in this report should be used to understand the risk exposure of the smart contract, project feasibility, and long-term sustainability, and as a guide to improving the smart contract's security posture by remediating the identified issues.

Tokenomics

▲ 5% tax when buying & selling

2% of trade goes to the marketing wallet in BNB
2% of trade distributed among holders in Golden coin
1% trade goes to the Liquidity Pool

Target market and the concept

- Anyone who’s interested in the Crypto space with long-term investment plans.
- Anyone who’s ready to earn a passive income by holding tokens.
- Anyone who’s interested in trading tokens.
- Anyone who’s interested in taking part in the Baby Golden Coin ecosystem.
- Anyone who’s interested in taking part in the future plans of Baby Golden Coin.
- Anyone who’s interested in making financial transactions with any other party Baby Golden Coin as the currency.

Potential to grow with score points

⚡ Project efficiency	8 / 10
🌟 Project uniqueness	8 / 10
📊 Information quality	8 / 10
👌 Service quality	8 / 10
💻 System quality	8 / 10
🌍 Impact on the community	8 / 10
💼 Impact on the business	9 / 10
🔮 Preparing for the future	8 / 10
🔒 Smart contract security	10 / 10
🛠️ Smart contract functionality assessment	10 / 10
🏆 **Total Score**	**8.5** / 10

Contract details

Token contract details for 15th of March 2024

Contract name	**Baby Golden Coin**
Contract address	**0x85cEf71A3277220cc3D88A1703Abc771cAA2241F**
Token supply	**1,000,000,000**
Token ticker	**BabyGolden**
Decimals	**9**
Token holders	**3**
Transaction count	**3**
Contract deployer address	**0x7b8fb187091B04e857d2454DFC46C8bD1823187E**
Contract’s current owner address	**0x89A77964246Ec5aefaBF10aECFb8b53ae9fbaC12**
Marketing address	**0x1BBa1335B6fE66929e3A33f3E66EC5A46d795753**
Dividend Tracker	**0x5Bf4224A55D6eCA5324696BA1ecB40fa24532392**

Contract code function details

№	Category	Item	Result
1	Coding conventions	ERC20 Token standards	PASS
		Compile errors	PASS
		Compiler version security	PASS
		Visibility specifiers	PASS
		Gas consumption	PASS
		SafeMath features	PASS
		Fallback usage	PASS
		tx.origin usage	PASS
		Deprecated items	PASS
		Redundant code	PASS
		Overriding variables	PASS
2	Function call audit	Authorization of function call	PASS
		Low level function (call/delegate call) security	PASS
		Returned value security	PASS
		Self destruct function security	PASS
3	Business security & centralisation	Access control of owners	PASS
		Business logics	PASS
		Business implementation	PASS
4	Integer overflow/underflow		PASS
5	Reentrancy		PASS
6	Exceptional reachable state		PASS
7	Transaction ordering dependence		PASS
8	Block properties dependence		PASS
9	Pseudo random number generator (PRNG)		PASS
10	DoS (Denial of Service)		PASS
11	Token vesting implementation		PASS
12	Fake deposit		PASS
13	Event security		PASS

Contract description table

The below table represents the summary of the contracts and methods in the token contract. We scanned the whole contract and listed down all the Interfaces, functions, and implementations with their visibility and mutability.

Contract	**Type**	**Bases**		
L	Function Name	Visibility	Mutability	Modifiers
BabyGolden	**Implementation**	**IERC20, Ownable**		
L		Public ❗	🛑	NO ❗
L		External ❗	💵	NO ❗
L	totalSupply	External ❗		NO ❗
L	name	Public ❗		NO ❗
L	symbol	Public ❗		NO ❗
L	decimals	Public ❗		NO ❗
L	balanceOf	Public ❗		NO ❗
L	getHolderDetails	Public ❗		NO ❗
L	getLastProcessedIndex	Public ❗		NO ❗
L	getNumberOfTokenHolders	Public ❗		NO ❗
L	totalDistributedRewards	Public ❗		NO ❗
L	allowance	External ❗		NO ❗
L	approve	Public ❗	🛑	NO ❗
L	_approve	Internal 🔒	🛑	
L	approveMax	External ❗	🛑	NO ❗
L	transfer	External ❗	🛑	NO ❗
L	transferFrom	External ❗	🛑	NO ❗
L	_transferFrom	Internal 🔒	🛑	
L	_basicTransfer	Internal 🔒	🛑	

└	shouldTakeFee	Internal 🔒		
└	takeFee	Internal 🔒	🛑	
└	shouldSwapBack	Internal 🔒		
└	clearStuckBalance	External ❗	🛑	onlyOwner
└	getBep20Tokens	External ❗	🛑	onlyOwner
└	whitelistPreSale	Public ❗	🛑	onlyOwner
└	___claimRewards	Public ❗	🛑	NO ❗
└	claimProcess	Public ❗	🛑	NO ❗
└	purgeRewards	External ❗	🛑	onlyOwner
└	isRewardExclude	Public ❗		NO ❗
└	isFeeExclude	Public ❗		NO ❗
└	swapBackInBnb	Internal 🔒	🛑	swapping
└	swapAndLiquify	Private 🔐	🛑	
└	swapTokensForEth	Private 🔐	🛑	
└	addLiquidity	Private 🔐	🛑	
└	setIsDividendExempt	External ❗	🛑	onlyOwner
└	setIsFeeExempt	External ❗	🛑	onlyOwner
└	setMarketingWallets	External ❗	🛑	onlyOwner
└	setSwapBackSettings	External ❗	🛑	onlyOwner
└	setDistributionCriteria	External ❗	🛑	onlyOwner
└	setDistributorSettings	External ❗	🛑	onlyOwner
Ownable	**Implementation**	**Context**		
└		Public ❗	🛑	NO ❗
└	owner	Public ❗		NO ❗
└	_checkOwner	Internal 🔒		
└	renounceOwnership	Public ❗	🛑	onlyOwner
└	transferOwnership	Public ❗	🛑	onlyOwner
└	_transferOwnership	Internal 🔒	🛑	

IERC20	**Interface**			
└	totalSupply	External ❗		NO ❗
└	balanceOf	External ❗		NO ❗
└	transfer	External ❗	🛑	NO ❗
└	allowance	External ❗		NO ❗
└	approve	External ❗	🛑	NO ❗
└	transferFrom	External ❗	🛑	NO ❗
Context	**Implementation**			
└	_msgSender	Internal 🔒		
└	_msgData	Internal 🔒		
└	_contextSuffixLength	Internal 🔒		
IUniswapV2 Factory	**Interface**			
└	feeTo	External ❗		NO ❗
└	feeToSetter	External ❗		NO ❗
└	getPair	External ❗		NO ❗
└	allPairs	External ❗		NO ❗
└	allPairsLength	External ❗		NO ❗
└	createPair	External ❗	🛑	NO ❗
└	setFeeTo	External ❗	🛑	NO ❗
└	setFeeToSetter	External ❗	🛑	NO ❗
IUniswapV2 Router01	**Interface**			
└	factory	External ❗		NO ❗
└	WETH	External ❗		NO ❗
└	addLiquidity	External ❗	🛑	NO ❗

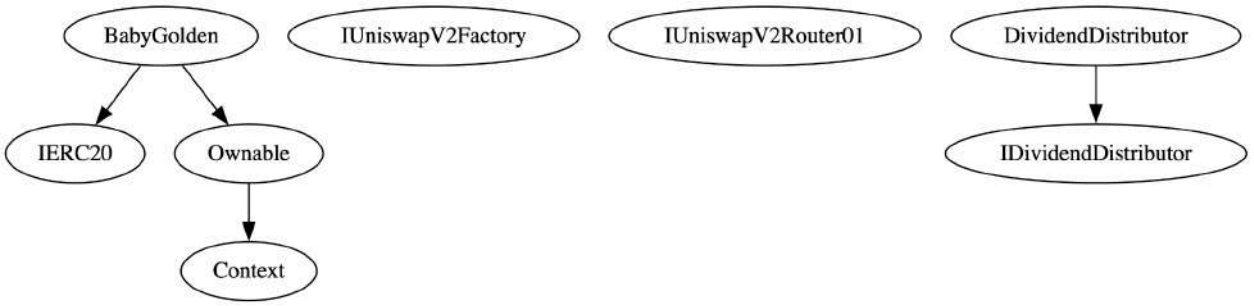
L	addLiquidityETH	External ❗	💵	NO ❗
L	removeLiquidity	External ❗	🛑	NO ❗
L	removeLiquidityETH	External ❗	🛑	NO ❗
L	removeLiquidityWithPermit	External ❗	🛑	NO ❗
L	removeLiquidityETHWithPermit	External ❗	🛑	NO ❗
L	swapExactTokensForTokens	External ❗	🛑	NO ❗
L	swapTokensForExactTokens	External ❗	🛑	NO ❗
L	swapExactETHForTokens	External ❗	💵	NO ❗
L	swapTokensForExactETH	External ❗	🛑	NO ❗
L	swapExactTokensForETH	External ❗	🛑	NO ❗
L	swapETHForExactTokens	External ❗	💵	NO ❗
L	quote	External ❗		NO ❗
L	getAmountOut	External ❗		NO ❗
L	getAmountIn	External ❗		NO ❗
L	getAmountsOut	External ❗		NO ❗
L	getAmountsIn	External ❗		NO ❗
IDividend Distributor	**Interface**			
L	setDistributionCriteria	External ❗	🛑	NO ❗
L	setShare	External ❗	🛑	NO ❗
L	deposit	External ❗	🛑	NO ❗
L	process	External ❗	🛑	NO ❗
L	purge	External ❗	🛑	NO ❗
Dividend Distributor	**Implementation**	**IDividend Distributor**		
L		Public ❗	🛑	NO ❗
L		External ❗	💵	NO ❗
L	setDistributionCriteria	External ❗	🛑	onlyToken

L	purge	External ❗	🛑	onlyToken
L	setShare	External ❗	🛑	onlyToken
L	deposit	External ❗	🛑	onlyToken
L	process	External ❗	🛑	onlyToken
L	shouldDistribute	Internal 🔒		
L	distributeDividend	Internal 🔒	🛑	
L	claimDividend	External ❗	🛑	NO ❗
L	claimDividendTo	External ❗	🛑	NO ❗
L	getUnpaidEarnings	Public ❗		NO ❗
L	getHolderDetails	Public ❗		NO ❗
L	getCumulativeDividends	Internal 🔒		
L	getLastProcessedIndex	External ❗		NO ❗
L	getNumberOfTokenHolders	External ❗		NO ❗
L	getShareHoldersList	External ❗		NO ❗
L	totalDistributedRewards	External ❗		NO ❗
L	addShareholder	Internal 🔒	🛑	
L	removeShareholder	Internal 🔒	🛑	
L	swapBnbForTokens	Private 🔐	🛑	

Legend

Symbol	Meaning
🛑	Function can modify state
💵	Function is payable

Inheritance Hierarchy



Security issue checking status

- High severity issues

  No low severity issues found

- Medium severity issues

  No medium severity issues found

- Low severity issues

  No high severity issues found

Owner privileges

- The owner can get contract BNB to owner wallet

```
function clearStuckBalance(uint256 amountPercentage) external onlyOwner {
    uint256 amountBNB = address(this).balance;
    payable(msg.sender).transfer((amountBNB * amountPercentage) / 100);
}
```

- The owner can withdraw any BEP20 tokens (can not withdraw native tokens)

```
function getBep20Tokens(
    address _tokenAddress,
    uint256 amount
) external onlyOwner {
    require(
        _tokenAddress != address(this),
        "You can not withdraw native tokens"
    );
    require(
        IERC20(_tokenAddress).balanceOf(address(this)) >= amount,
        "No Enough Tokens"
    );
    IERC20(_tokenAddress).transfer(msg.sender, amount);
}
```

- The owner can whitelist the pre-sale address from the contract

```
function whitelistPreSale(address _preSale) public onlyOwner {
    isFeeExempt[_preSale] = true;
    isDividendExempt[_preSale] = true;
}
```

❖ The owner can withdraw reward tokens from the reward tracker

```
function purgeRewards() external onlyOwner {
    dividendTracker.purge(msg.sender);
}
```

❖ The owner can include/exclude wallets from rewards

```
function setIsDividendExempt(
    address holder,
    bool exempt
) external onlyOwner {
    require(holder != address(this) && holder != pair);
    isDividendExempt[holder] = exempt;
    if (exempt) {
        dividendTracker.setShare(holder, 0);
    } else {
        dividendTracker.setShare(holder, _balances[holder]);
    }
}
```

❖ The owner can include/exclude wallets from fees

```
function setIsFeeExempt(address holder, bool exempt) external onlyOwner {
    isFeeExempt[holder] = exempt;
}
```

❖ The owner can change the marketing wallet

```
function setMarketingWallets(address _newWallet) external onlyOwner {
    marketingWallet = _newWallet;
}
```

❖ The owner can enable/disable swapping and can change the swap threshold

```
function setSwapBackSettings(
    bool _enabled,
    uint256 _amount
) external onlyOwner {
    swapEnabled = _enabled;
    swapThreshold = _amount;
}
```

❖ The owner can change the distribution criteria

```
function setDistributionCriteria(
    uint256 _minPeriod,
    uint256 _minDistribution
) external onlyOwner {
    dividendTracker.setDistributionCriteria(_minPeriod, _minDistribution);
}
```

❖ The owner can change max gas to use when processing rewards tracker

```
function setDistributorSettings(uint256 gas) external onlyOwner {
    require(gas < 750000);
    distributorGas = gas;
}
```

Audit conclusion

RugFreeCoins team has performed in-depth testing, line-by-line manual code review, and automated audit of the smart contract. The smart contract was analyzed mainly for common smart contract vulnerabilities, exploits, manipulations, and hacks. According to the smart contract audit.

Smart contract functional Status:	PASS
Smart contract security Status:	NONE
Number of risk issues:	NONE
Solidity code functional issue level:	PASS
Number of owner privileges:	10
Centralization risk correlated to the active owner:	HIGH
Smart contract active ownership:	ACTIVE