RugFreeCoins Audit

BabyCMC Token Audit

Smart Contract Security Audit

August 26, 2021

Contents

Audit details

Audited project

BabyCMC Token

Contract Address

0xc460c98b39e24b4984af73ad38ddfb577d6c3ff2

Client contact

BabyCMC Token Team

Blockchain

Binance smart chain

Project website

https://www.babycmc.io/

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 BabyCMC to perform an audit of the smart contract.

https://bscscan.com/token/0xc460c98b39e24b4984af73ad38ddfb577d6c3ff2

The focus of this audit is to verify 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, long term sustainability and as a guide to improve the security posture of the smart contract by remediating the issues that were identified.

About the project

BabyCMC is a unique project built on Binance smart chain. CMC isn't a coin, but a coin listing platform. Every day, a few lucky coins across the blockchain get listed on CoinMarketCap (“CMC”). It is no secret that every project in DeFi that aspires to the moon must go through the CMC listing process. It is also very well known that often, getting listed on CMC means an immediate moonshot for the lucky investors who get in exactly at the time of listing.

Every day, BabyCMC screens through the most recently listed coins on CMC and distributes rewards in the most promising one. BabyCMS, each transaction, purchase and sales incur a 22% fee.

Tokenomics

- ❖ 18% of trade goes to holders pockets in different tokens listed on CMC as per the BabyCMC’s decision every single day.
- ❖ 3% of trade goes to development, marketing and research.
- ❖ 1% of trade goes to the liquidity pool.

Roadmap

Phase 1

- Creation of Community and Website.
- Creation of Socials.
- Organic Growth.
- Dev Wallet Locked.
- 50% of Supply Burned.
- Presale and LP Lock (by DxSale).
- Audit by RugFreeCoins.

Phase 2

- Whitelist Contest for Head Start in Presale and Anti-Bot Measures.
- Most Promising CMC Recently Listed Coin Announced for Rewards.
- PCS Launch.
- Start of Trending #1 on DexTools.
- Coingecko fast listing.
- Twitter-based influencer marketing push.

Phase 3

- CMC fast listing.
- CMC and Dextools trending.
- Referral and Shilling Contests.
- Implementation of new reward selection strategy.
- Manual BuyBacks and Burns.
- Listing on an exchange.

Target market and the concept

Target market

- Anyone who's interested in Crypto space with long term investment plans.
- Anyone who's ready to build a completely different portfolio with different tokens by holding BabyCMC tokens.
- Anyone who's interested in trading tokens.
- Anyone who's interested in taking part in BabyCMC's future plans.
- Anyone who's interested in making financial transactions with any other party using BabyCMC as the currency.

Core concept

CoinMarketCap is the reference for all the crypto investors in the world looking for sophisticated projects. The goal of BabyCMC is to find opportunities with a 2 to 5x short term potential to benefit our investors. The goal is to transform BabyCMC’s 18% rewards to a 30% to 90% return on rewards through the short-term appreciation of the token rewarded. This way, BabyCMC investors would only pay 22% in total transaction fees to potentially receive a much higher percentage in rewards, thanks to the price increase of the coin selected by the BabyCMC team.

Every day, BabyCMC screens through the most recently listed coins on CMC and distributes rewards in the most promising one.

BabyCMC team considers multiple aspects when selecting coins with the highest short term potential.

- Overall project concept
- Investor protection - this includes external audits but also our own vetting process
- Development team expertise and whether they are doxxed
- Time since launch
- Market Capitalization, Volume and Liquidity Ratios
- Chart trends
- Audience targets
- Time of listing and geographies awake at time of listing

The reward system

18% of each transaction when buying and selling will be deducted and sent amongst all holders from one of the tokens that’s going to be listed on CMC. Every day, the BabyCMC team decides which token will be given as rewards by screening the token's feasibility and potential. The rewards are sent to holders that have at least 1 BabyCMC tokens, holders will be eligible to receive rewards proportional to how many tokens each individual holds.

Sustainable mechanism

The **fee of 3% marketing, development and research** is what allows BabyCMC to use them to promote the token and use funds to further development of the platform. Tokens will be swapped into BNBs and will be sent to a marketing wallet per transaction. This way, BabyCMC’s will have access to the funds without selling tokens as the traditional way, which will enable them to consume funds without hurting the project.

The liquidity fee of 1%, which is a redistribution mechanism that ensures the trading pool always has sufficient liquidity.

Anti-bot measures

No single wallet will be allowed to purchase more than 0.5% of the total supply in one transaction. This restriction, coupled with a 50 second cooldown period, will limit bot users from fluctuating the price at launch or at any period in the life of the token.

White-listed presale

To prevent bots from snatching all the tokens, wallets will be whitelisted to take part with the presale.

Anti-whale measures

No single private wallet will be allowed to own more than 1% of the total supply to avoid large holders having too much pricing power over the rest of the community.

Potential to grow with score points

1.	Project efficiency	9/10
2.	Project uniqueness	10/10
3	Information quality	10/10
4	Service quality	9/10
5	System quality	9/10
6	Impact on the community	9/10
7	Impact on the business	9/10
8	Preparing for the future	8/10
Total Points		**9.13/10**

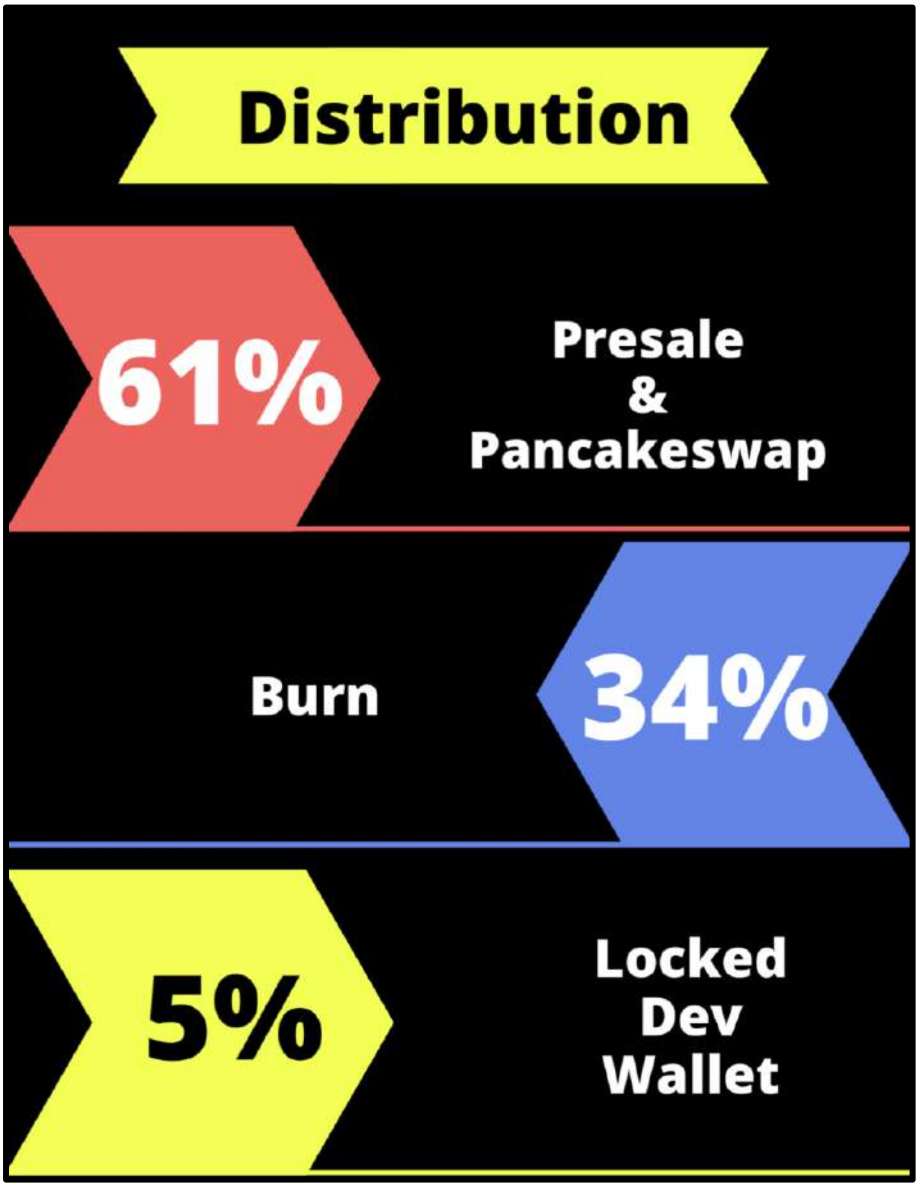
Contract details

Token contract details for 26th August 2021

Contract name	BabyCMC Token
Contract address	0xc460c98b39e24b4984af73ad38ddfb577d6c3ff2
Token supply	1,000,000,000
Token ticker	CMC
Decimals	9
Token holders	3
Transaction count	4
Distributor address	0xfa8d49f3c166c4cd61653280b7835c2031f0ee93
Marketing address	Not publicly visible
Contract deployer address	0x0B15A7575796f0DEF049A1fE295678E94AA9530e
Contract's current owner address	0x0b15a7575796f0def049a1fe295678e94aa9530e

Token distribution

Tokens are distributed as follows:



Contract code function details

No	Category	Item	Result
1	Coding conventions	BRC20 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
		Selfdestruct function security	pass
3	Business security	Access control of owners	pass
		Business logics	pass
		Business implementations	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

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 its visibility and mutability.

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 ❗		NO❗
└	decimals	External ❗		NO❗
└	symbol	External ❗		NO❗
└	name	External ❗		NO❗
└	getOwner	External ❗		NO❗
└	balanceOf	External ❗		NO❗
└	transfer	External ❗	🛑	NO❗
└	allowance	External ❗		NO❗
└	approve	External ❗	🛑	NO❗
└	transferFrom	External ❗	🛑	NO❗
Auth	**Implementation**			
└		Public ❗	🛑	NO❗

└	authorize	Public ❗️	🛑	onlyOwner
└	unauthorize	Public ❗️	🛑	onlyOwner
└	isOwner	Public ❗️		NO❗️
└	isAuthorized	Public ❗️		NO❗️
└	transferOwnership	Public ❗️	🛑	onlyOwner
IDEXFactory	**Interface**			
└	createPair	External ❗️	🛑	NO❗️
IDEXRouter	**Interface**			
└	factory	External ❗️		NO❗️
└	WETH	External ❗️		NO❗️
└	addLiquidity	External ❗️	🛑	NO❗️
└	addLiquidityETH	External ❗️	💵	NO❗️
└	swapExactTokensForTokensSupportingFeeOnTransferTokens	External ❗️	🛑	NO❗️
└	swapExactETHForTokensSupportingFeeOnTransferTokens	External ❗️	💵	NO❗️
└	swapExactTokensForETHSupportingFeeOnTransferTokens	External ❗️	🛑	NO❗️
IDividendDistributor	**Interface**			
└	setDistributionCriteria	External ❗️	🛑	NO❗️
└	setShare	External ❗️	🛑	NO❗️
└	deposit	External ❗️	💵	NO❗️
└	process	External ❗️	🛑	NO❗️
└	purge	External ❗️	🛑	NO❗️

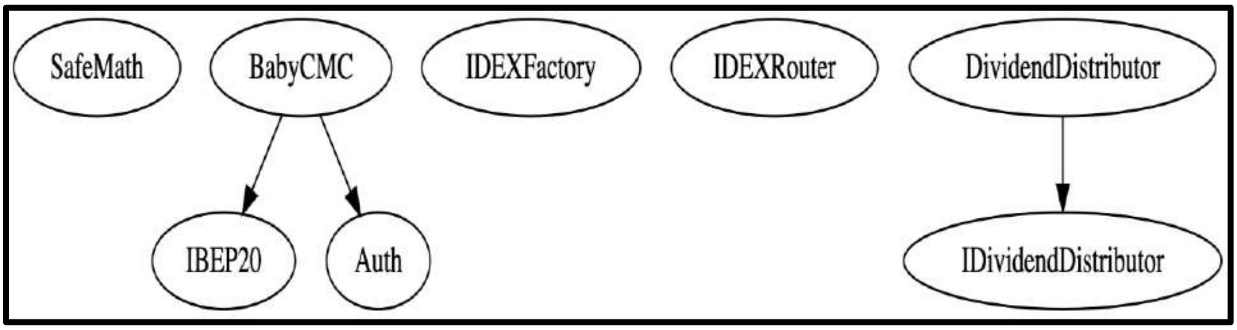
DividendDistributor	**Implementation**	**IDividendDistributor**		
└		Public ❗	🛑	NO❗
└	setDistributionCriteria	External ❗	🛑	onlyToken
└	purge	External ❗	🛑	onlyToken
└	setShare	External ❗	🛑	onlyToken
└	deposit	External ❗	💵	onlyToken
└	process	External ❗	🛑	onlyToken
└	shouldDistribute	Internal 🔒		
└	distributeDividend	Internal 🔒	🛑	
└	claimDividend	External ❗	🛑	NO❗
└	getUnpaidEarnings	Public ❗		NO❗
└	getCumulativeDividends	Internal 🔒		
└	addShareholder	Internal 🔒	🛑	
└	removeShareholder	Internal 🔒	🛑	
BabyCMC	**Implementation**	**IBEP20, Auth**		
└		Public ❗	🛑	Auth
└		External ❗	💵	NO❗
└	totalSupply	External ❗		NO❗
└	decimals	External ❗		NO❗
└	symbol	External ❗		NO❗
└	name	External ❗		NO❗
└	getOwner	External ❗		NO❗
└	balanceOf	Public ❗		NO❗
└	allowance	External ❗		NO❗
└	approve	Public ❗	🛑	NO❗
└	approveMax	External ❗	🛑	NO❗
└	transfer	External ❗	🛑	NO❗
└	transferFrom	External ❗	🛑	NO❗

└	setMaxWalletPercent	External ❗	🛑	onlyOwner
└	_transferFrom	Internal 🔒	🛑	
└	_basicTransfer	Internal 🔒	🛑	
└	checkTxLimit	Internal 🔒		
└	shouldTakeFee	Internal 🔒		
└	takeFee	Internal 🔒	🛑	
└	shouldSwapBack	Internal 🔒		
└	clearStuckBalance	External ❗	🛑	onlyOwner
└	tradingStatus	Public ❗	🛑	onlyOwner
└	cooldownEnabled	Public ❗	🛑	onlyOwner
└	purgeBeforeSwitch	Public ❗	🛑	onlyOwner
└	switchToken	Public ❗	🛑	onlyOwner
└	___claimRewards	Public ❗	🛑	NO❗
└	claimProcess	Public ❗	🛑	NO❗
└	swapBack	Internal 🔒	🛑	swapping
└	setTxLimit	External ❗	🛑	authorized
└	setIsDividendExempt	External ❗	🛑	authorized
└	setIsFeeExempt	External ❗	🛑	authorized
└	setIsTxLimitExempt	External ❗	🛑	authorized
└	setIsTimelockExempt	External ❗	🛑	authorized
└	setFees	External ❗	🛑	authorized
└	setFeeReceivers	External ❗	🛑	authorized
└	setSwapBackSettings	External ❗	🛑	authorized
└	setTargetLiquidity	External ❗	🛑	authorized
└	setDistributionCriteria	External ❗	🛑	authorized
└	setDistributorSettings	External ❗	🛑	authorized
└	getCirculatingSupply	Public ❗		NO❗
└	getLiquidityBacking	Public ❗		NO❗
└	isOverLiquified	Public ❗		NO❗
└	claimTokens	External ❗	🛑	onlyOwner
└	airdropFixed	External ❗	🛑	onlyOwner

Legend

Symbol	Meaning
🛑	**Function can modify state**
💵	**Function is payable**

Inheritance Hierarchy

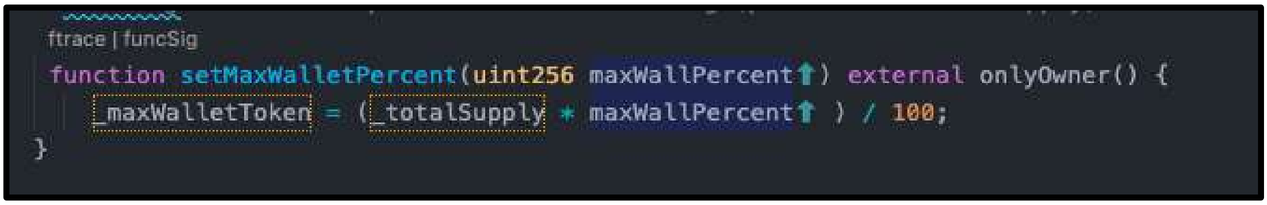


Security issue checking status

- **High severity issues**
  - **No high severity issues found.**
- **Medium severity issues**
  - **No medium severity issues found.**
- **Low severity issues**
  - **No low severity issues found.**

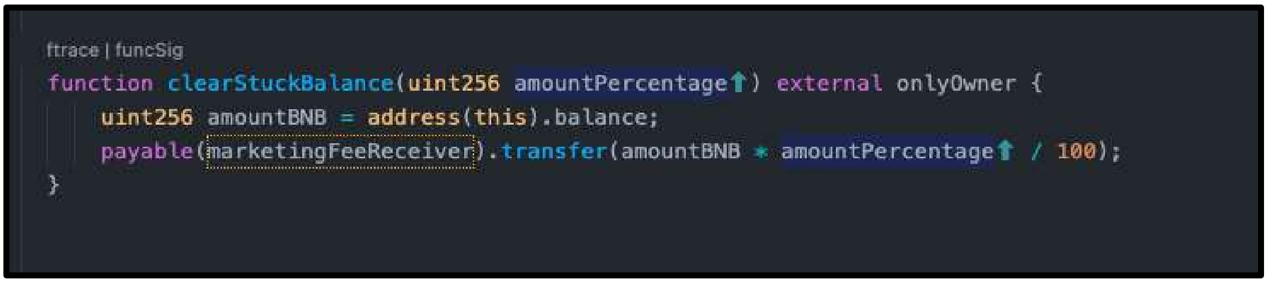
Owner privileges

❖ The owner can change the max wallet percentage.



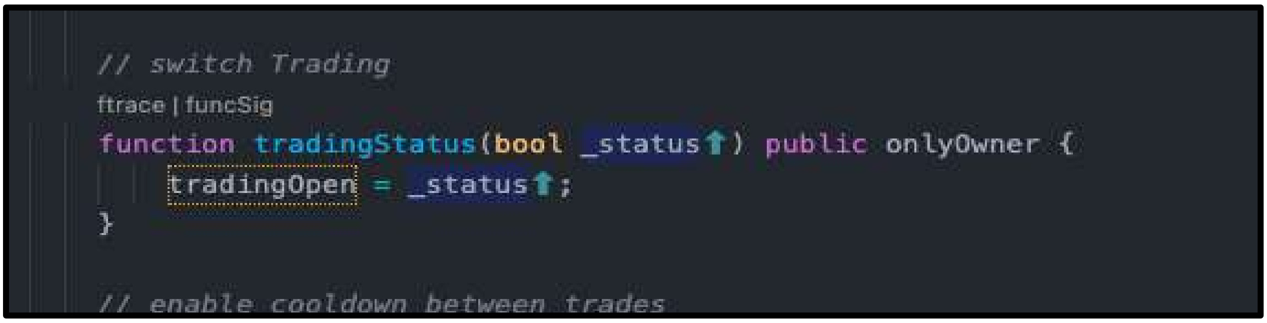
```
ftrace | funcSig
function setMaxWalletPercent(uint256 maxWallPercent) external onlyOwner() {
    _maxWalletToken = (_totalSupply * maxWallPercent ) / 100;
}
```

❖ The owner can transfer the contract BNB balance to marketing wallet.



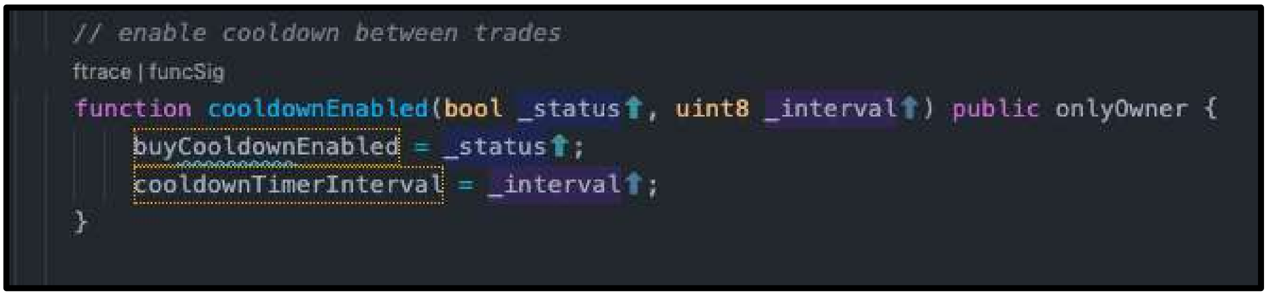
```
ftrace | funcSig
function clearStuckBalance(uint256 amountPercentage) external onlyOwner {
    uint256 amountBNB = address(this).balance;
    payable(marketingFeeReceiver).transfer(amountBNB * amountPercentage / 100);
}
```

❖ The owner can enable and disable trading.



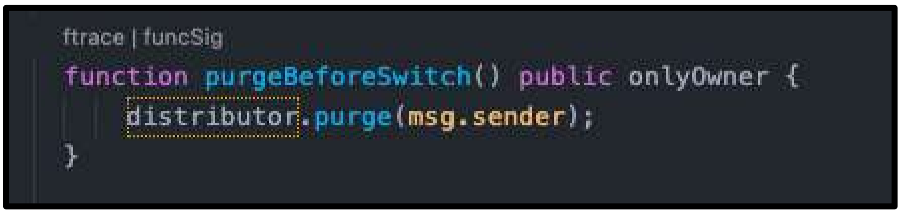
```
// switch Trading
ftrace | funcSig
function tradingStatus(bool _status) public onlyOwner {
    tradingOpen = _status;
}

// enable cooldown between trades
```

❖ The owner can enable/disable and change buy cool down time.



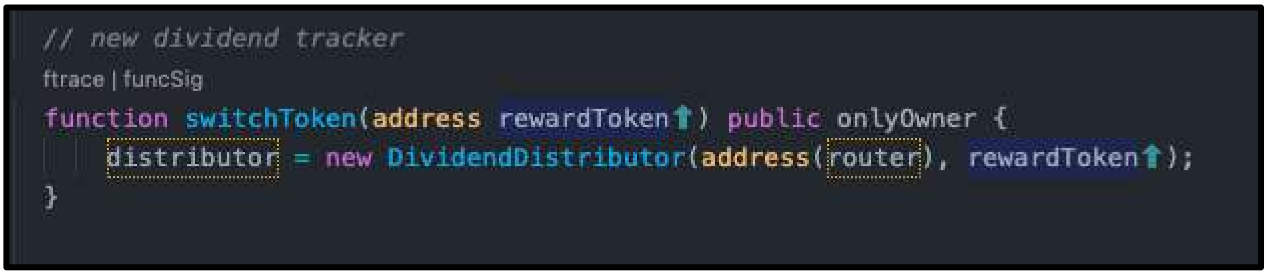
```
// enable cooldown between trades
ftrace | funcSig
function cooldownEnabled(bool _status, uint8 _interval) public onlyOwner {
    buyCooldownEnabled = _status;
    cooldownTimerInterval = _interval;
}
```

- The owner can clear the balance from dividend tracker.



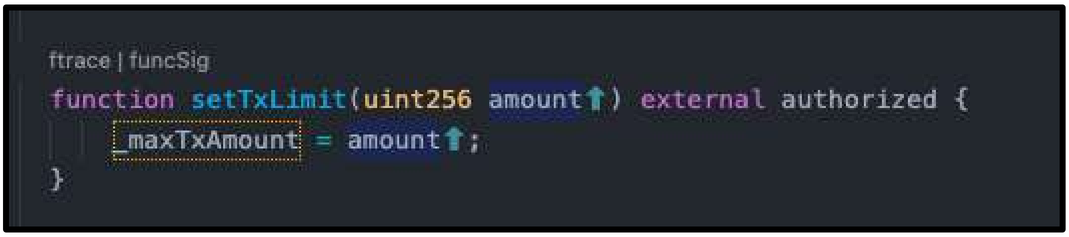
```
ftrace | funcSig
function purgeBeforeSwitch() public onlyOwner {
    distributor.purge(msg.sender);
}
```

- The owner can change the distribution token.



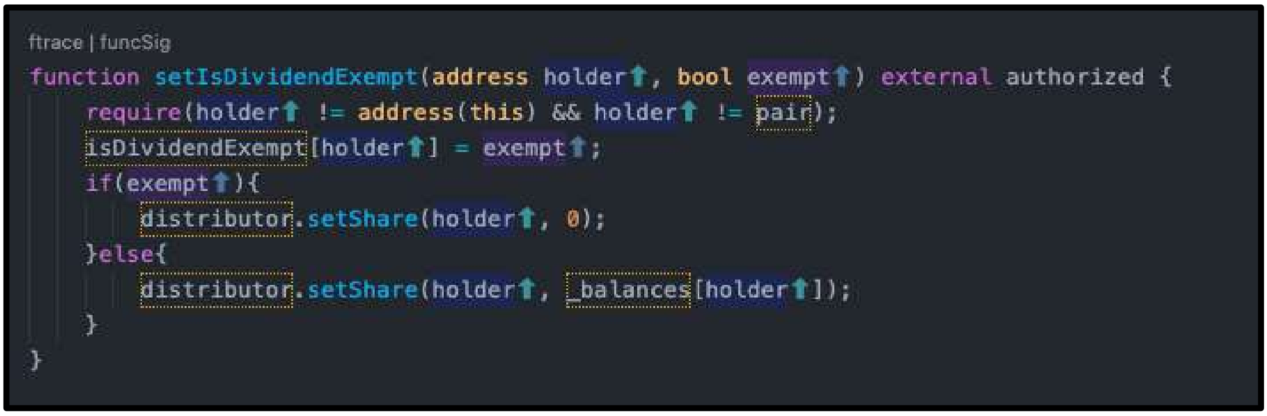
```
// new dividend tracker
ftrace | funcSig
function switchToken(address rewardToken) public onlyOwner {
    distributor = new DividendDistributor(address(router), rewardToken);
}
```

- The owner can change the tax limit.



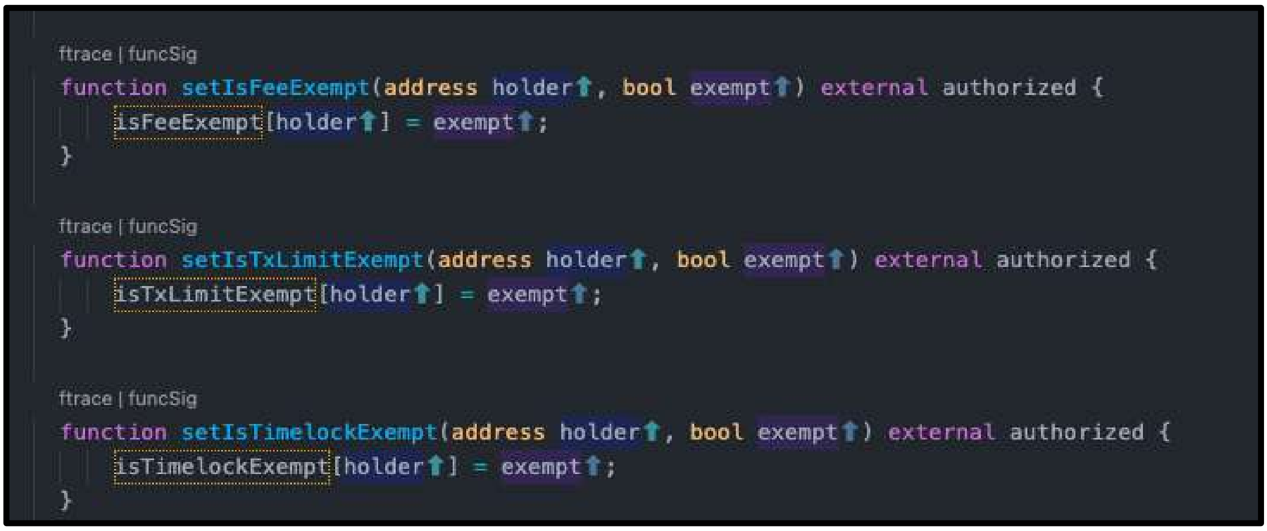
```
ftrace | funcSig
function setTxLimit(uint256 amount) external authorized {
    _maxTxAmount = amount;
}
```

- The owner can exempt wallets from dividend.



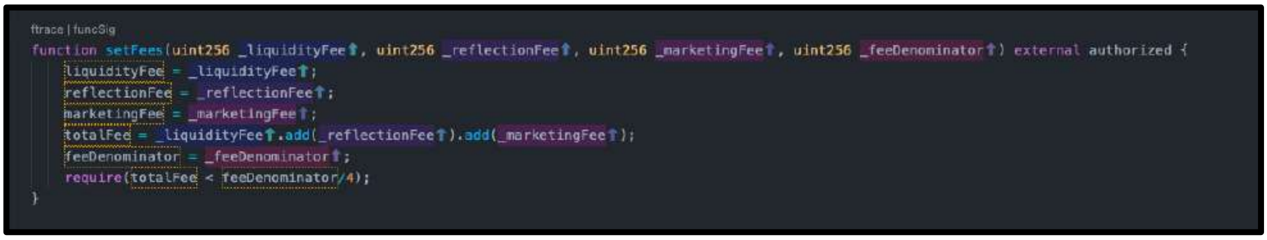
```
ftrace | funcSig
function setIsDividendExempt(address holder, bool exempt) external authorized {
    require(holder != address(this) && holder != pair);
    isDividendExempt[holder] = exempt;
    if(exempt){
        distributor.setShare(holder, 0);
    }else{
        distributor.setShare(holder,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

- The owner can exempt wallets from fees, max transaction and time lock.



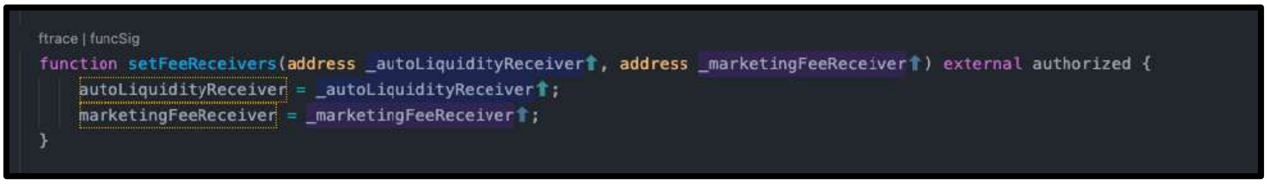
```
ftrace | funcSig
function setIsFeeExempt(address holder, bool exempt) external authorized {
    isFeeExempt[holder] = exempt;
}

ftrace | funcSig
function setIsTxLimitExempt(address holder, bool exempt) external authorized {
    isTxLimitExempt[holder] = exempt;
}

ftrace | funcSig
function setIsTimelockExempt(address holder, bool exempt) external authorized {
    isTimelockExempt[holder] = exempt;
}
```

- The owner can change fees.



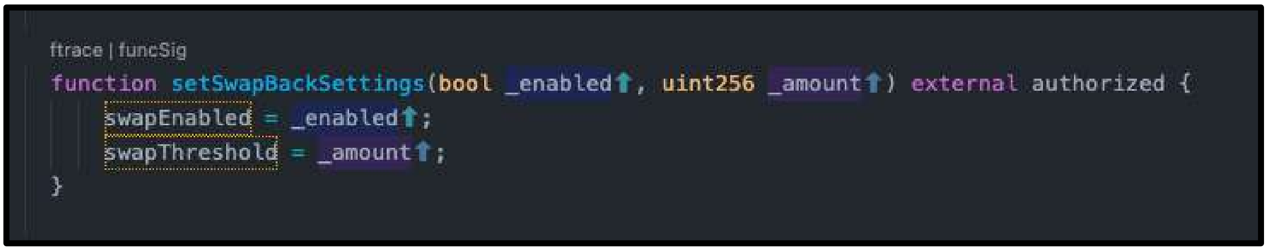
```
ftrace | funcSig
function setFees(uint256 _liquidityFee, uint256 _reflectionFee, uint256 _marketingFee, uint256 _feeDenominator) external authorized {
    liquidityFee = _liquidityFee;
    reflectionFee = _reflectionFee;
    marketingFee = _marketingFee;
    totalFee = _liquidityFee.add(_reflectionFee).add(_marketingFee);
    feeDenominator = _feeDenominator;
    require(totalFee < feeDenominator/4);
}
```

- The owner can change the marketing wallet and liquidity wallet.



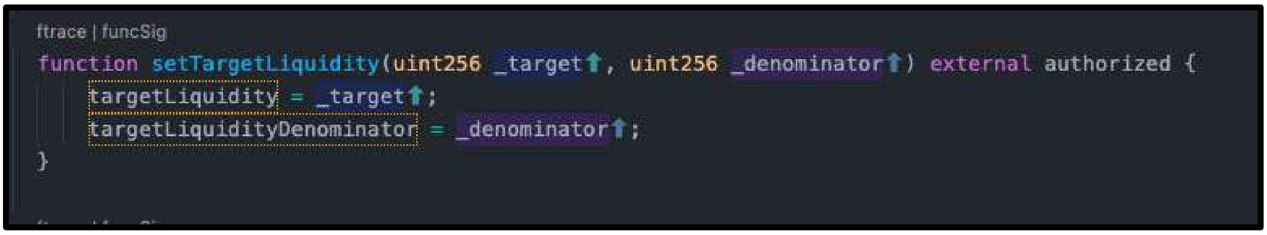
```
ftrace | funcSig
function setFeeReceivers(address _autoLiquidityReceiver, address _marketingFeeReceiver) external authorized {
    autoLiquidityReceiver = _autoLiquidityReceiver;
    marketingFeeReceiver = _marketingFeeReceiver;
}
```

- The owner can change swap back settings.



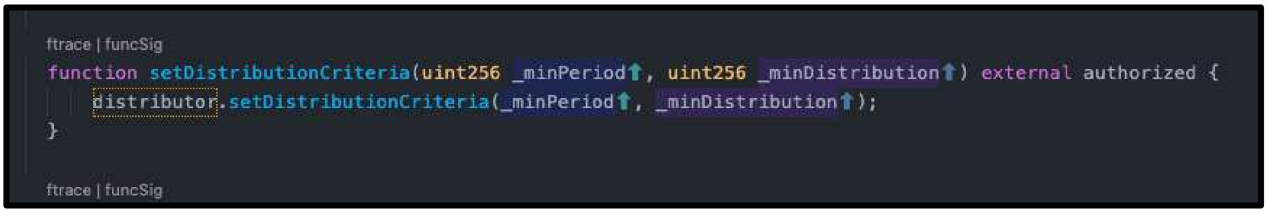
```
ftrace | funcSig
function setSwapBackSettings(bool _enabled, uint256 _amount) external authorized {
    swapEnabled = _enabled;
    swapThreshold = _amount;
}
```

- The owner can change the target liquidity amount.



```
ftrace | funcSig
function setTargetLiquidity(uint256 _target, uint256 _denominator) external authorized {
    targetLiquidity = _target;
    targetLiquidityDenominator = _denominator;
}
```

- ❖ The owner can change the minimum distribution time and distribution amount.



```
ftrace | funcSig
function setDistributionCriteria(uint256 _minPeriod, uint256 _minDistribution) external authorized {
    distributor.setDistributionCriteria(_minPeriod, _minDistribution);
}

ftrace | funcSig
```

Audit conclusion

While conducting the audit of the BabyCMC smart contract, it was observed that there is nothing alarming with the code.