CheckPoint

Token Security Audit Report
Prepared for BabyFOMO

[v.1.0]

September 2021

Document Properties

Client	BabyFOMO
Platform	Binance Smart Chain
Language	Solidity
Codebase	0x7c6Daa947Ae509c5842485cDc96bcD1bdAA11449

Audit Summary

Delivery Date	08.09.2021
Audit Methodology	Static Analysis, Manual Review
Auditor(s)	Erno Patiala
Classification	Publlic
Version	1.0

Contact Information

Company	CheckPoint
Name	Hanna Järvinen
Telegram	t.me/checkpointreport
E-mail	contact@checkpoint.report

Remark: For more information about this document and its contents, please contact CheckPoint team

Table Of Contents

1 Executive Summary

On 08/09/2021, CheckPoint conducted a full audit for the BabyFOMO to verify the overall security posture including a smart contract review to discover issues and vulnerabilities in the source code. Static Code Analysis, Dynamic Analysis, and Manual Review werdone in conjunction to identify smart contract vulnerabilities together with technical & business logic flaws that may be exposed to the potential risk of the platform and the ecosystem.

After further analysis and internal discussion, we determined a few issues of varying severities that need to be brought up and paid more attention to. More information can be found in **Section 5 'Audit Result'**. Practical recommendations are provided according to each vulnerability found and should be followed to remediate the issue.

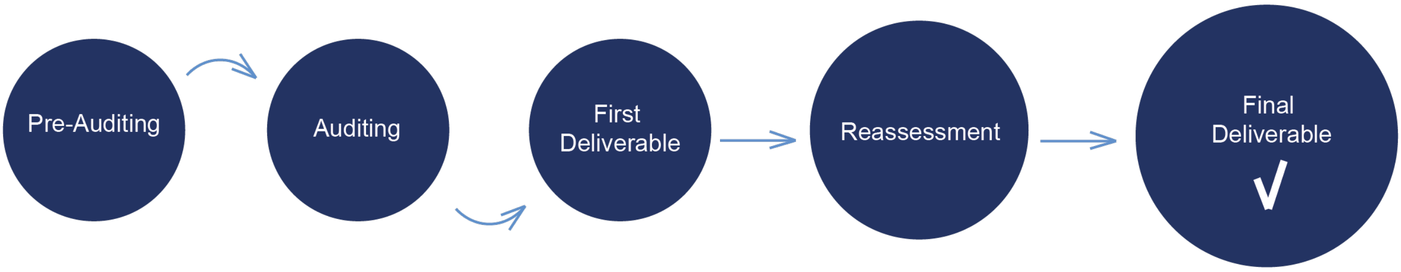
🟥

BabyFOMO
High Risk Level

Communication Channels	Website Content Analysis, Social Media Listening
Smart Contract Code	Smart Contract Details, Contract Function Details, Issues Checking Status, Detailed Findings Information

THIS TOKEN PASSES CHECKPOINT'S SECURITY VERIFICATION STANDART

2 Audit Methodology



CheckPoint conducts the following procedure to enhance the security level of our clients' tokens:

- **Pre-Auditing**

  Planning a comprehensive survey of the token, its ecosystem, possible risks & prospects, getting to understand the overall operations of the related smart contracts, checking for readiness, and preparing for the auditing.

- **Auditing**

  Study of all available information about the token on the Web, inspecting the smart contracts using automated analysis tools and manual analysis by a team of professionals.

- **First Deliverable and Consulting**

  Delivering a preliminary report on the findings with suggestions on how to remediate those issues and providing consultation.

- **Reassessment**

  Verifying the status of the issues and whether there are any other complications in the fixes applied.

- **Final Deliverable**

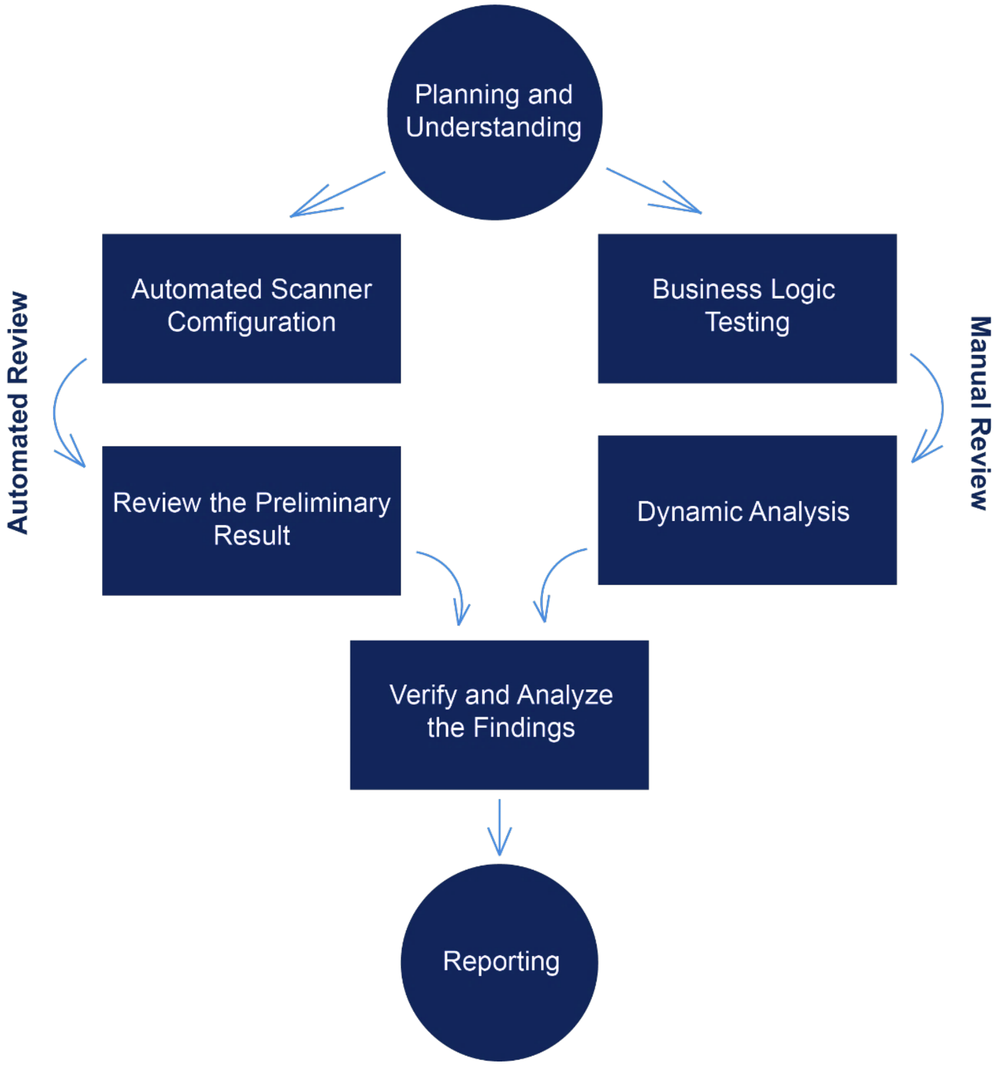
  Providing a full report with the detailed status of each issue.

The security audit process of CheckPoint includes three types testing:

1. Examining publicly available information about the token on social networks, including a detailed overview of the official website and analysis of the latest messages and opinions about the token.

2. Smart contract codes are scanned/tested for commonly known and more specific vulnerabilities using automated analysis tools.

3. Manual audit of the codes for security issues. The contracts are manually analyzed to look for any potential problems.


Planning and Understanding
Automated Scanner Comfiguration
Business Logic Testing
Automated Review
Manual Review
Review the Preliminary Result
Dynamic Analysis
Verify and Analyze the Findings
Reporting

Remark: Manual and Automated review approaches can be mixed and matched including business logic analysis in terms of malicious doers' perspective

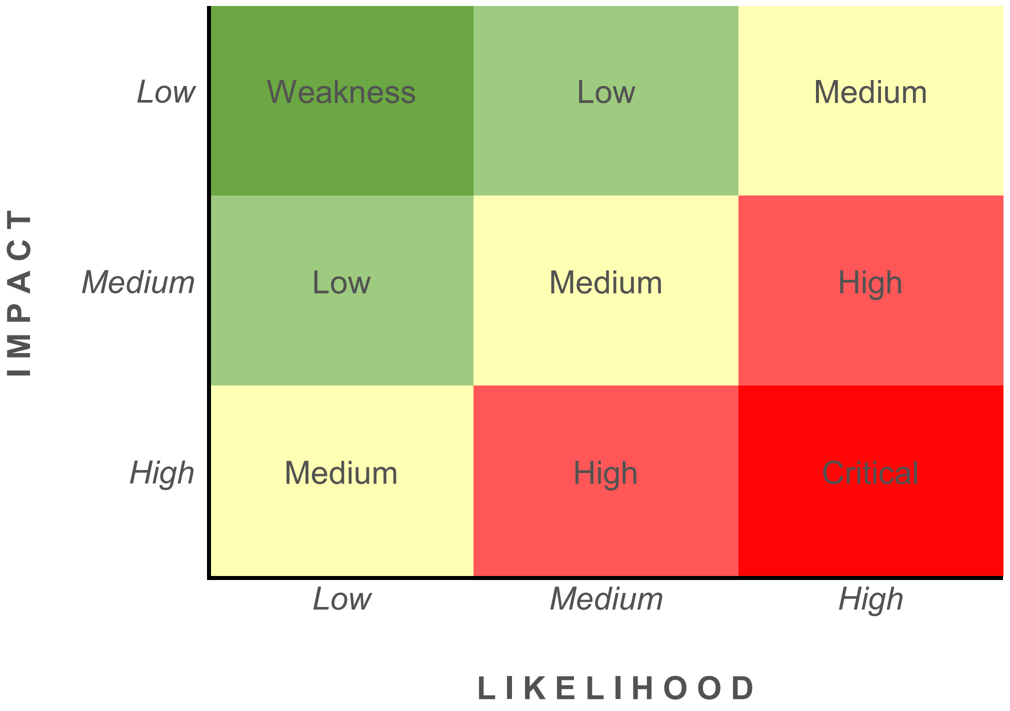
In particular, we perform the audit according to the following procedure:

- **Planning & Understanding**
    - determine scope of testing and understand application purpose and workflows;
    - identify key risk areas, including technical and business risks;
    - determine approach – which sections to review within the resource constraints and review method – automated, manual or mixed.
- **Automated Review**
    - adjust automated source code review tools to inspect the code for known unsafe coding patterns;
    - verify output of the tool in order to eliminate false positive result, and if necessary, adjust and re-run the code review tool.
- **Manual Review**
    - testing for business logic flaws requires thinking in unconventional methods;
    - identify unsafe coding behavior via static code analysis.
- **Reporting**
    - analyze the root cause of the flaws;
    - recommend coding process improvements.

3 Risk Level Classification

To standardize the evaluation, we define the following terminology based on OWASP Risk Rating Methodology:

- **Likelihood** represents how likely a particular vulnerability is to be uncovered and exploited in the wild.
- **Impact** measures the technical loss and business damage of a successful attack.
- **Severity** demonstrates the overall criticality of the risk and calculated as the product of impact and likelihood values, illustrated in a twodimensional matrix. The shading of the matrix visualizes the different risk levels.



IMPACT	Low	Medium	High
Low	Weakness	Low	Medium
Medium	Low	Medium	High
High	Medium	High	Critical

LIKELIHOOD

Remark: Likelihood and Impact are categorized into three levels: H, M, and L, i.e., High, Medium and Low respectively. Severity is determined by likelihood and impact and can be classified into five categories accordingly, i.e., Critical, High, Medium, Low and Weakness

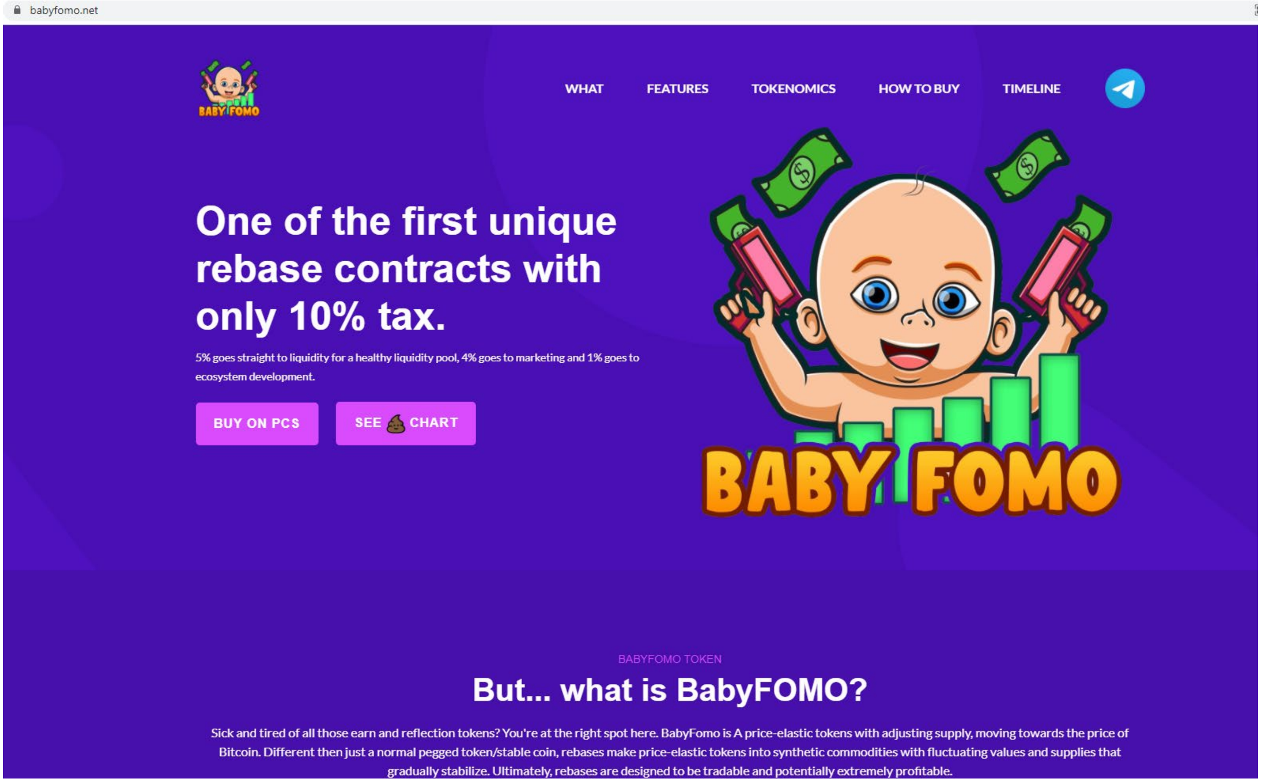
For prioritization of the vulnerabilities, we have adopted the scheme by five distinct levels for risk: Critical, High, Medium, Low, and Weakness. The risk level definitions are presented in table.

LEVEL	DESCRIPTION
Weakness	There are safety risks theoretically, but it is extremely difficult to reproduce in engineering
Low	Low severity vulnerabilities may affect the operation of the DeFi project in certain scenarios
Medium	Medium severity vulnerability will affect the operation of the DeFi project. It is recommended to fix medium-risk vulnerabilities
High	High severity vulnerabilities will affect the normal operation of the DeFi project. It is strongly recommended to fix high-risk vulnerabilities
Critical	Critical severity vulnerabilities will have a significant impact on the security of the DeFi project

4 Project Overview

4.1 Communication Channels

https://babyfomo.net/



One of the first unique rebase contracts with only 10% tax.

5% goes straight to liquidity for a healthy liquidity pool, 4% goes to marketing and 1% goes to ecosystem development.

But... what is BabyFOMO?

Sick and tired of all those earn and reflection tokens? You're at the right spot here. BabyFomo is A price-elastic tokens with adjusting supply, moving towards the price of Bitcoin. Different then just a normal pegged token/stable coin, rebases make price-elastic tokens into synthetic commodities with fluctuating values and supplies that gradually stabilize. Ultimately, rebases are designed to be tradable and potentially extremely profitable.

Website was registered on 02-09-2021, registration expires 02-09-2022.

Above the image is an actual snapshot of the current live website of the project.

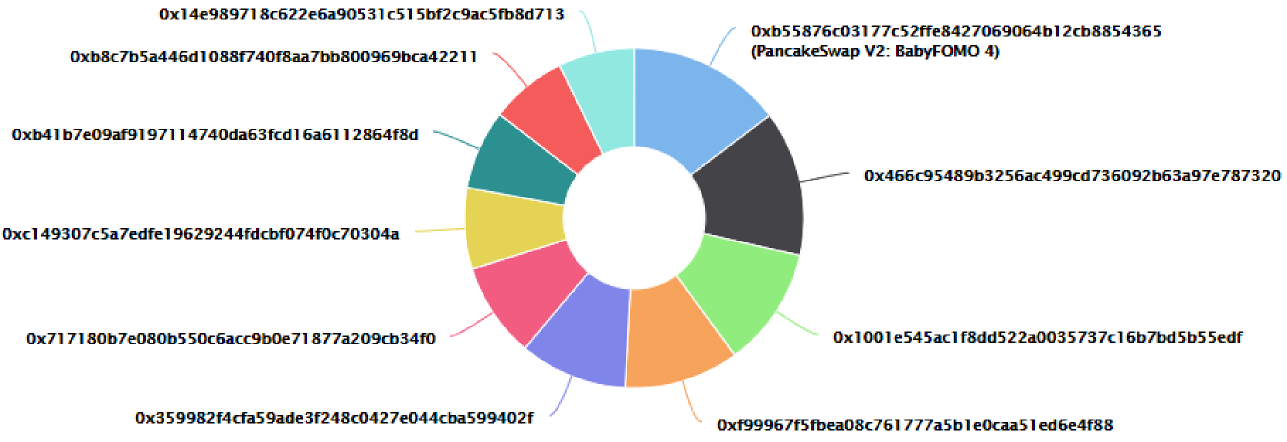
- ✓ **Mobile Friendly**
- ✓ **No JavaScript Errors**
- ✓ **Visionary Roadmap**
- ✓ **Spell Check**
- ✓ **Valid SSL Certificate**
- ✓ **2 Social Media Networks**
- ✓ **1000+ Telegram Members**
- ✓ **100+ Twitter Followers**
- ✓ **Active voice chats**
- ✓ **No injected spam and popus found**

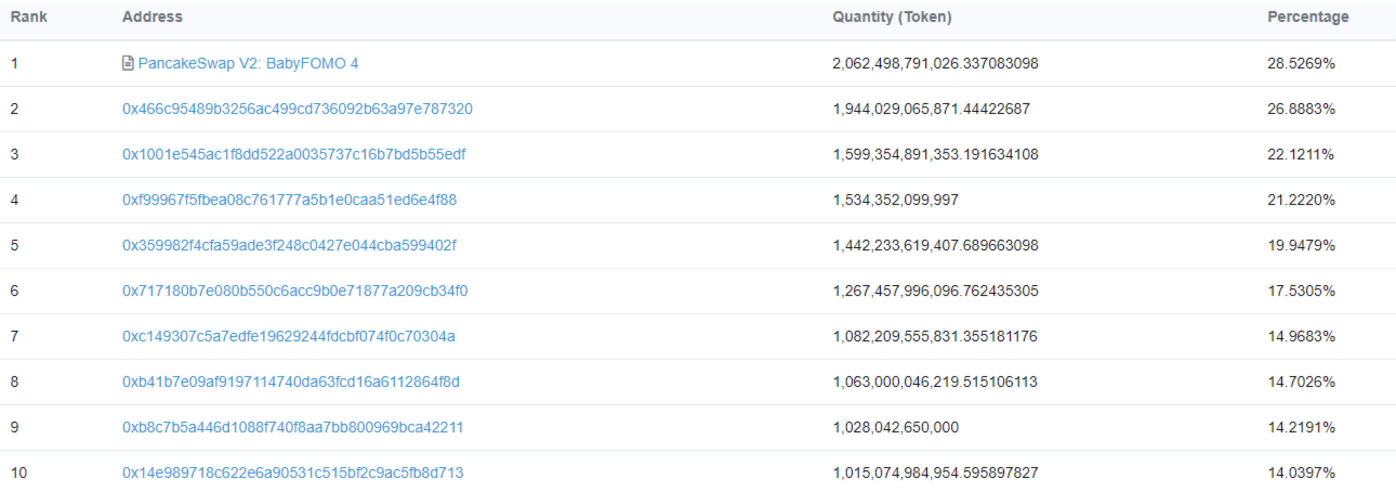
Remark: This page contains active links

4.2 Smart Contract Details

Contract Name	BabyFOMO
Contract Address	0x7c6Daa947Ae509c5842485cDc96bcD1bdAA11449
Total Supply	7,230,012,889,280.143983 **[RISK]**
Token Ticker	BabyFOMO
Decimals	9
Token Holders	514
Transactions Count	5,282
Top 10 Holders Dominance	472,23%
Liquidity Fee	5%
Marketing Fee	4%
Ecosystem Fee	2%
Total Fee	11%
Pair Contract	0xb55876c03177c52ffe8427069064b12cb8854365
Contract Deployer Address	0xd39f894a2681d259044cc70779388e77f9c5c426
Current Owner Address	0xd39f894a2681d259044cc70779388e77f9c5c426

BabyFOMO Top 10 Token Holders



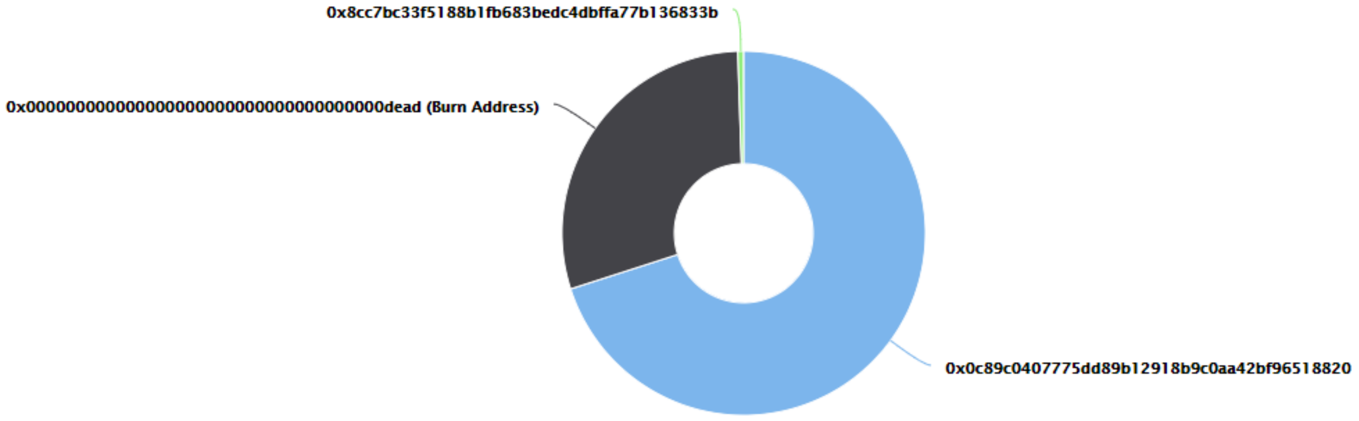


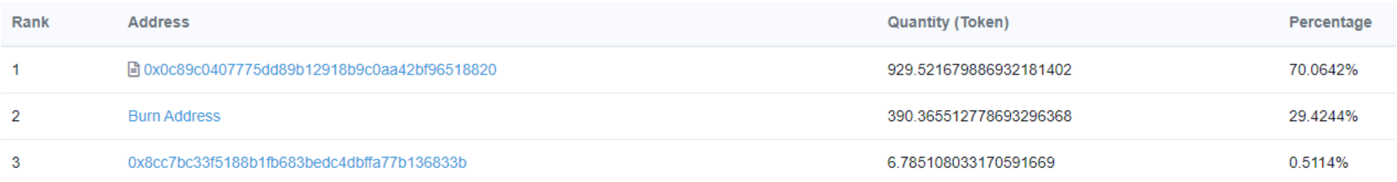
Rank	Address	Quantity (Token)	Percentage
1	PancakeSwap V2: BabyFOMO 4	2,062,498,791,026.337083098	28.5269%
2	0x466c95489b3256ac499cd736092b63a97e787320	1,944,029,065,871.44422687	26.8883%
3	0x1001e545ac1f8dd522a0035737c16b7bd5b55edf	1,599,354,891,353.191634108	22.1211%
4	0xf99967f5fbea08c761777a5b1e0caa51ed6e4f88	1,534,352,099,997	21.2220%
5	0x359982f4cfa59ade3f248c0427e044cba599402f	1,442,233,619,407.689663098	19.9479%
6	0x717180b7e080b550c6acc9b0e71877a209cb34f0	1,267,457,996,096.762435305	17.5305%
7	0xc149307c5a7edfe19629244fdcbf074f0c70304a	1,082,209,555,831.355181176	14.9683%
8	0xb41b7e09af9197114740da63fcd16a6112864f8d	1,063,000,046,219.515106113	14.7026%
9	0xb8c7b5a446d1088f740f8aa7bb800969bca42211	1,028,042,650,000	14.2191%
10	0x14e989718c622e6a90531c515bf2c9ac5fb8d713	1,015,074,984,954.595897827	14.0397%

✓ **PancakeSwap holds 28,5% of the token's supply as liquidity**

[RISK] 10 top holders have more $BabyFOMO than 100% total supply

BabyFOMO Top 3 LP Token Holders

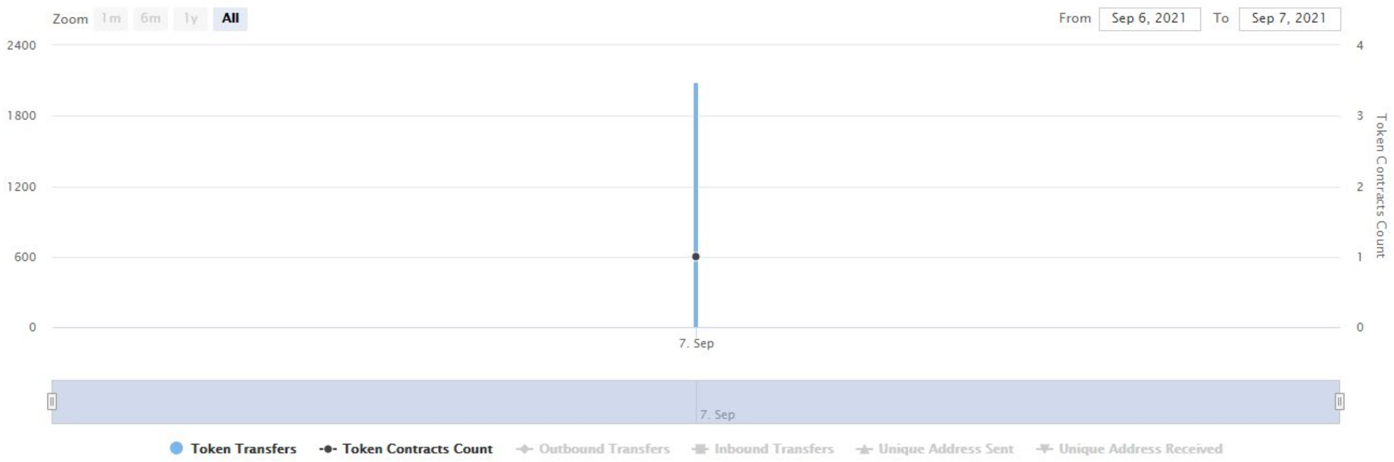




Rank	Address	Quantity (Token)	Percentage
1	0x0c89c0407775dd89b12918b9c0aa42bf96518820	929.521679886932181402	70.0642%
2	Burn Address	390.365512778693296368	29.4244%
3	0x8cc7bc33f5188b1fb683bedc4dbffa77b136833b	6.785108033170591669	0.5114%

✓ ~29% LP tokens are permanently removed from circulation

BabyFOMO Contract Interaction Details



4.4 Issues Checking Status

CHECKING ITEM	NOTES	RESULT
Arbitrary Jump with Function Type Variable	N / A	PASS
Arithmetic Accuracy Deviation	N / A	PASS
Assert Violation	N / A	PASS
Authorization through tx.origin	N / A	PASS
Business Logic	N / A	HIGH RISK
Code with No Effects	N / A	PASS
Critical Solidity Compiler	N / A	PASS
Delegatecall to Untrusted Callee	N / A	PASS
Design Logic	N / A	PASS
DoS with Block Gas Limit	N / A	LOW RISK
DoS with Failed Call	N / A	PASS
Function Default Visibility	N / A	PASS
Hash Collisions With MVLA	N / A	PASS
Incorrect Constructor Name	N / A	PASS
Incorrect Inheritance Order	N / A	PASS
Integer Overflows and Underflows	N / A	PASS
Lack of Proper Signature Verification	N / A	PASS
Message Call with Hardcoded Gas Amount	N / A	PASS
Missing Protection Against SRA	N / A	PASS
Presence of Unused Variables	N / A	PASS
Reentrancy	N / A	PASS
Requirement Violation	N / A	PASS

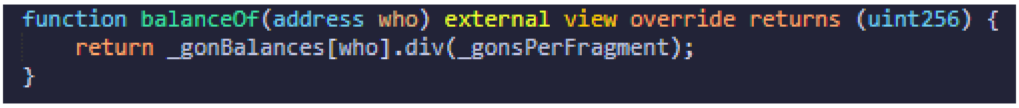
CHECKING ITEM	NOTES	RESULT
Right-To-Left-Override Control Character	N / A	PASS
Shadowing State Variables	N / A	PASS
Signature Malleability	N / A	PASS
State Variable Default Visibility	N / A	PASS
Timestamp Dependence	N / A	PASS
Transaction Order Dependence	N / A	PASS
Typographical Error	N / A	PASS
Unencrypted Private Data On-Chain	N / A	PASS
Unexpected Ether balance	N / A	PASS
Uninitialized Storage Pointer	N / A	PASS
Use of Deprecated Solidity Functions	N / A	PASS
Weak Sources of Randomness From CA	N / A	PASS
Write to Arbitrary Storage Location	N / A	PASS

Remark: To evaluate the risk, we go through a list of check items and each would be labeled with a severity category. For one check item, if our tool or analysis does not identify any issue, the contract is considered safe regarding the check item

4.5 Detailed Findings Information

[RISK] Business Logic

- totalSupply() output doesn't match the real total supply.

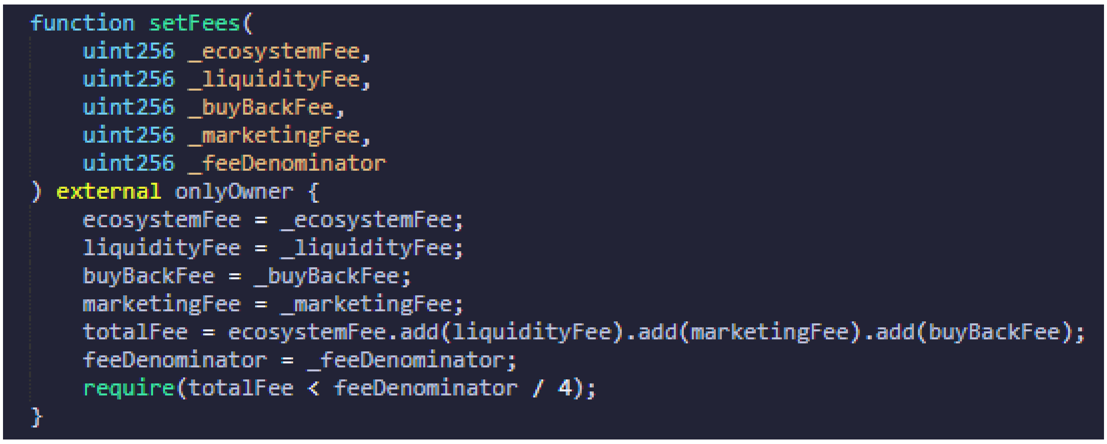


```
function balanceOf(address who) external view override returns (uint256) {
    return _gonBalances[who].div(_gonsPerFragment);
}
```

[RISK] Owner Privileges (in the period when the owner is not renounced)

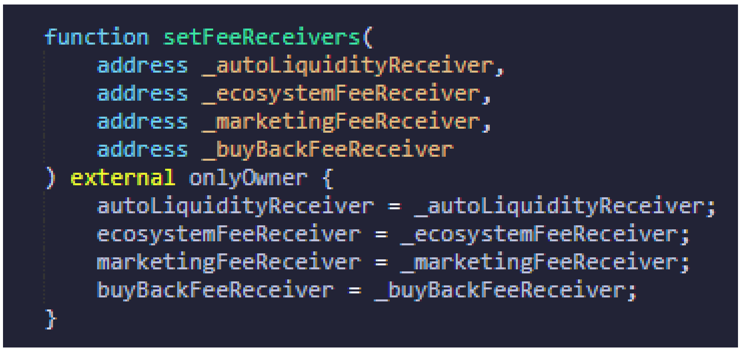
The contract contains the following privileged functions that are restricted by the onlyOwner.

- The owner can change ecosystem fee, liquidity fee, buyback fee and marketing fee.



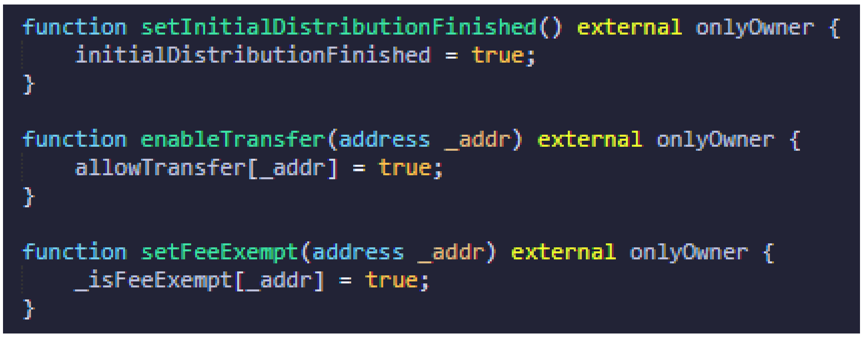
```
function setFees(
    uint256 _ecosystemFee,
    uint256 _liquidityFee,
    uint256 _buyBackFee,
    uint256 _marketingFee,
    uint256 _feeDenominator
) external onlyOwner {
    ecosystemFee = _ecosystemFee;
    liquidityFee = _liquidityFee;
    buyBackFee = _buyBackFee;
    marketingFee = _marketingFee;
    totalFee = ecosystemFee.add(liquidityFee).add(marketingFee).add(buyBackFee);
    feeDenominator = _feeDenominator;
    require(totalFee < feeDenominator / 4);
}
```

- The owner can set fee recievers.



```
function setFeeReceivers(
    address _autoLiquidityReceiver,
    address _ecosystemFeeReceiver,
    address _marketingFeeReceiver,
    address _buyBackFeeReceiver
) external onlyOwner {
    autoLiquidityReceiver = _autoLiquidityReceiver;
    ecosystemFeeReceiver = _ecosystemFeeReceiver;
    marketingFeeReceiver = _marketingFeeReceiver;
    buyBackFeeReceiver = _buyBackFeeReceiver;
}
```

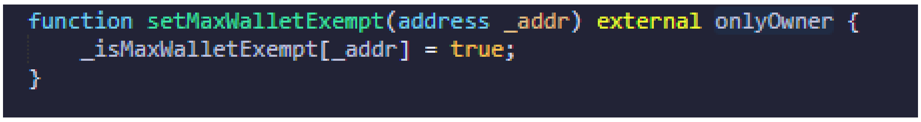
- The owner can set fee exempt and max wallet token.



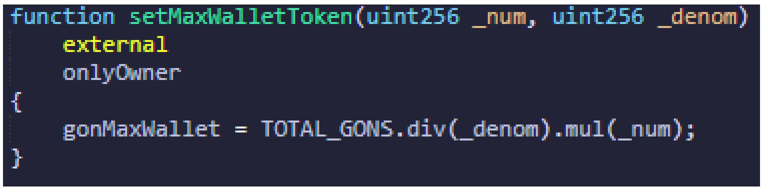
```
function setInitialDistributionFinished() external onlyOwner {
    initialDistributionFinished = true;
}

function enableTransfer(address _addr) external onlyOwner {
    allowTransfer[_addr] = true;
}

function setFeeExempt(address _addr) external onlyOwner {
    _isFeeExempt[_addr] = true;
}
```



```
function setMaxWalletExempt(address _addr) external onlyOwner {
    _isMaxWalletExempt[_addr] = true;
}
```



```
function setMaxWalletToken(uint256 _num, uint256 _denom)
    external
    onlyOwner
{
    gonMaxWallet = TOTAL_GONS.div(_denom).mul(_num);
}
```

5 Audit Result

LEVEL	ISSUES
High	**Business Logic (1)**
Medium	**Owner Privilegies (3)**

1. The contract utilizes SafeMath libraries along with following the ERC20 standard.

2. The contract implements the function balanceOf, the caller can this function to query token balance. However, the token balance changes with ‘_gonsPerFragment’, that is, the actual queried token balance is based on the rounding of ‘_gonsPerFragment’ (the total user balance is different with the total supply).

3. There is a 'Marketing fee', a 'Liquidity fee', an ‘Ecosystem Fee’ and a ‘Buyback Fee’ on all transactions for any non-excluded address that participates in a transfer. The owner has the ability to modify these to any percentage fees at any time.

4. The owner has the ability to set and update a maximum transaction percent at any time, which will impose a limit to the number of tokens that can be transferred during any given transaction.

5. This maximum transaction amount does not apply to the owner during transactions where the owner is either the sender or the recipient.

5.1 Findings Summary

🟥 **BabyFOMO**

High Risk Level

- ✓ No external vulnerabilities were identified within the smart contract's code
- ✓ We strongly recommend that the team renounces ownership
- ✓ Please ensure trust in the team prior to investing as they have substantial control within the ecosystem
- ✓ We strongly recommend that the contract owners remove errors and re-audit

6 Disclamer

CheckPoint team issues this report with reference to the facts that have occurred or existed before the issuance of this report, and only assumes corresponding responsibility based on these. For the facts that occurred or existed after the issuance, CheckPoint is not able to judge the security status of this project, and is not responsible for them. The security audit analysis and other contents of this report are based on the documents and materials provided to CheckPoint by the information provider till the date of the insurance report. CheckPoint is not responsible for the background and other conditions of the project.

This security audit is not produced to supplant any other type of assessment and does not guarantee the discovery of all security vulnerabilities within the scope of the assessment. However, we warrant that this audit is conducted with goodwill, professional approach, and competence. Since an assessment from one single party cannot be confirmed to cover all possible issues within the smart contract(s), CheckPoint suggests conducting multiple independent assessments to minimize the risks. Lastly, nothing contained in this audit report should be considered as investment advice.

CheckPoint

Website

https://checkpoint.report

E-mail

contact@checkpoint.report

Telegram

@checkpointreport

Github

https://github.com/checkpointreport