



Building the Futuristic **Blockchain Ecosystem**

SECURITY AUDIT REPORT

MAGGIESIMPSON

TOKEN OVERVIEW

Risk Findings

Severity	Found
● High	0
● Medium	1
● Low	0
● Informational	0

Centralization Risks

Owner Privileges	Description
● Can Owner Set Taxes >25% ?	Not Detected
● Owner needs to enable trading ?	Not Detected
● Can Owner Disable Trades ?	Not Detected
● Can Owner Mint ?	Not Detected
● Can Owner Blacklist ?	Not Detected
● Can Owner set Max Wallet amount ?	Not Detected
● Can Owner Set Max TX amount ?	Not Detected

TABLE OF CONTENTS

02	Token Overview	
03	Table of Contents	
04	Overview	
05	Contract Details	
06	Audit Methodology	
07	Vulnerabilities Checklist	
08	Risk Classification	
09	Inheritance Trees	
10	Function Details	
13	Testnet Version	
15	Manual Review	
17	About Expelee	
18	Disclaimer	

OVERVIEW

The Expelee team has performed a line-by-line manual analysis and automated review of the smart contract. The smart contract was analysed mainly for common smart contract vulnerabilities, exploits, and manipulation hacks. According to the smart contract audit:

Audit Result	Passed
KYC Verification	-
Audit Date	16 June 2023

CONTRACT DETAILS

Token Name: MaggieSimpson

Symbol: Maggie

Network: Binance Smart Chain

Language: Solidity

Contract Address:

0x90f0Ef7261138152A656c6d014f3fc5A689DAdFA

Total Supply: 420,000,000,000,000

Owner's Wallet:

0xd0099CC1D1FC0D272478E363B9B0316C9180a903

Deployer's Wallet:

0xd0099CC1D1FC0D272478E363B9B0316C9180a903

Testnet.

<https://testnet.bscscan.com/address/0xfabC5c382C3D20eFE97Ed77Bec8f2e4613f5caac#code>

AUDIT METHODOLOGY

Audit Details

Our comprehensive audit report provides a full overview of the audited system's architecture, smart contract codebase, and details on any vulnerabilities found within the system.

Audit Goals

The audit goal is to ensure that the project is built to protect investors and users, preventing potentially catastrophic vulnerabilities after launch, that lead to scams and rugpulls.

Code Quality

Our analysis includes both automatic tests and manual code analysis for the following aspects:

- Exploits
- Back-doors
- Vulnerability
- Accuracy
- Readability

Tools

- DE
- Open Zeppelin
- Code Analyzer
- Solidity Code
- Compiler
- Hardhat

VULNERABILITY CHECKS

Design Logic	Passed
Compiler warnings	Passed
Private user data leaks	Passed
Timestamps dependence	Passed
Integer overflow and underflow	Passed
Race conditions & reentrancy. Cross-function race conditions	Passed
Possible delays in data delivery	Passed
Oracle calls	Passed
Front Running	Passed
DoS with Revert	Passed
DoS with block gas limit	Passed
Methods execution permissions	Passed
Economy model	Passed
Impact of the exchange rate on the logic	Passed
Malicious event log	Passed
Scoping and declarations	Passed
Uninitialized storage pointers	Passed
Arithmetic accuracy	Passed
Cross-function race conditions	Passed
Safe Zepplin module	Passed

RISK CLASSIFICATION

When performing smart contract audits, our specialists look for known vulnerabilities as well as logical and access control issues within the code. The exploitation of these issues by malicious actors may cause serious financial damage to projects that failed to get an audit in time. We categorize these vulnerabilities by the following levels:

High Risk

Issues on this level are critical to the smart contract's performance/functionality and should be fixed before moving to a live environment.

Medium Risk

Issues on this level are critical to the smart contract's performance/functionality and should be fixed before moving to a live environment.

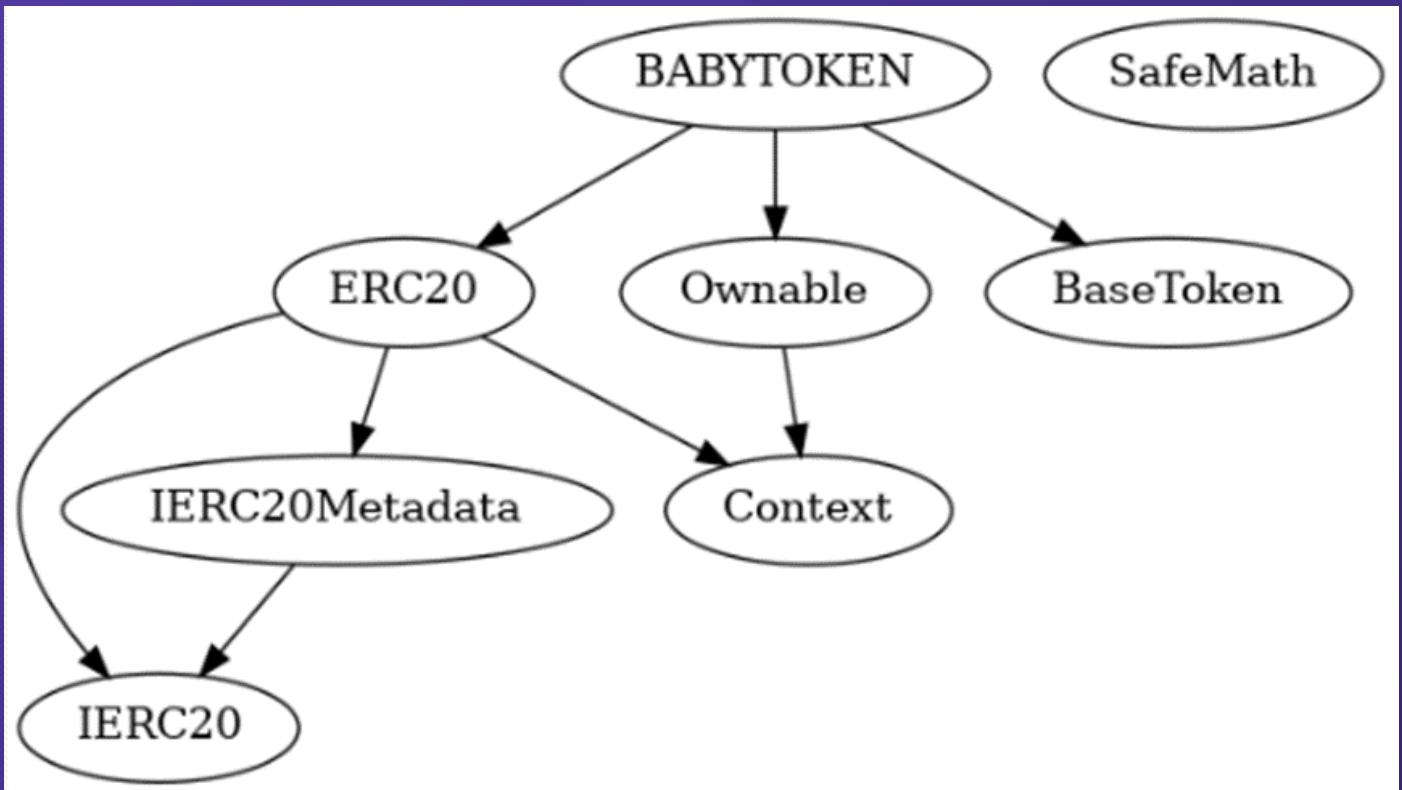
Low Risk

Issues on this level are minor details and warnings that can remain unfixed.

Informational

Issues on this level are minor details and warnings that can remain unfixed.

INHERITANCE TREES



FUNCTION DETAILS

Contract	Type	Bases			
----- ----- ----- ----- -----					
L	**Function Name**	**Visibility**	**Mutability**	**Modifiers**	
IERC20 Interface					
L	totalSupply	External	!	NO !	
L	balanceOf	External	!	NO !	
L	transfer	External	!	NO !	
L	allowance	External	!	NO !	
L	approve	External	!	NO !	
L	transferFrom	External	!	NO !	
IERC20Metadata Interface IERC20					
L	name	External	!	NO !	
L	symbol	External	!	NO !	
L	decimals	External	!	NO !	
Context Implementation					
L	_msgSender	Internal	🔒		
L	_msgData	Internal	🔒		
ERC20 Implementation Context, IERC20, IERC20Metadata					
L	<Constructor>	Public	!	NO !	
L	name	Public	!	NO !	
L	symbol	Public	!	NO !	
L	decimals	Public	!	NO !	
L	totalSupply	Public	!	NO !	
L	balanceOf	Public	!	NO !	
L	transfer	Public	!	NO !	
L	allowance	Public	!	NO !	
L	approve	Public	!	NO !	
L	transferFrom	Public	!	NO !	
L	increaseAllowance	Public	!	NO !	
L	decreaseAllowance	Public	!	NO !	
L	_transfer	Internal	🔒		
L	_mint	Internal	🔒		
L	_burn	Internal	🔒		
L	_approve	Internal	🔒		
L	_beforeTokenTransfer	Internal	🔒		
L	_afterTokenTransfer	Internal	🔒		
Ownable Implementation Context					
L	<Constructor>	Public	!	NO !	
L	owner	Public	!	NO !	
L	renounceOwnership	Public	!	onlyOwner	
L	transferOwnership	Public	!	onlyOwner	
L	_setOwner	Private	🔒		

FUNCTION DETAILS

```

||||| | | |
| **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 | 🔒 | ||
|||||
| **Clones** | Library | |||
| | clone | Internal | 🔒 | ● | ||
| | cloneDeterministic | Internal | 🔒 | ● | ||
| | predictDeterministicAddress | Internal | 🔒 | ||
| | predictDeterministicAddress | Internal | 🔒 | ||
|||||
| **Address** | Library | |||
| | isContract | Internal | 🔒 | ||
| | sendValue | Internal | 🔒 | ● | ||
| | functionCall | Internal | 🔒 | ● | ||
| | functionCall | Internal | 🔒 | ● | ||
| | functionCallWithValue | Internal | 🔒 | ● | ||
| | functionCallWithValue | Internal | 🔒 | ● | ||
| | functionStaticCall | Internal | 🔒 | ||
| | functionStaticCall | Internal | 🔒 | ||
| | functionDelegateCall | Internal | 🔒 | ● | ||
| | functionDelegateCall | Internal | 🔒 | ● | ||
| | verifyCallResult | Internal | 🔒 | ||
|||||
| **IUniswapV2Factory** | 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 ! |
|||||
| **IUniswapV2Router01** | Interface | |||

```

FUNCTION DETAILS

```

| L | factory | External ! | | NO ! |
| L | WETH | External ! | | NO ! |
| L | 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 ! |
|||||
| **IUniswapV2Router02** | Interface | IUniswapV2Router01 |||
| L | removeLiquidityETHSupportingFeeOnTransferTokens | External ! | ● | NO ! |
| L | removeLiquidityETHWithPermitSupportingFeeOnTransferTokens | External ! | ● | NO ! |
|
| L | swapExactTokensForTokensSupportingFeeOnTransferTokens | External ! | ● | NO ! |
| L | swapExactETHForTokensSupportingFeeOnTransferTokens | External ! | 🟢 | NO ! |
| L | swapExactTokensForETHSupportingFeeOnTransferTokens | External ! | ● | NO ! |
|||||
| **IERC20Upgradeable** | Interface | |||
| L | totalSupply | External ! | | NO ! |
| L | balanceOf | External ! | | NO ! |
| L | transfer | External ! | ● | NO ! |
| L | allowance | External ! | | NO ! |
| L | approve | External ! | ● | NO ! |
| L | transferFrom | External ! | ● | NO ! |
|||||
| **IERC20MetadataUpgradeable** | Interface | IERC20Upgradeable |||
| L | name | External ! | | NO ! |
| L | symbol | External ! | | NO ! |
| L | decimals | External ! | | NO ! |
|||||
| **Initializable** | Implementation | |||
|||||
| **ContextUpgradeable** | Implementation | Initializable |||
| L | __Context_init | Internal 🔒 | ● | initializer |
| L | __Context_init_unchained | Internal 🔒 | ● | initializer |
| L | _msgSender | Internal 🔒 | | |

```

FUNCTION DETAILS

```

| L | _msgData | Internal | 🔒 | | |
|||||
| **ERC20Upgradeable** | Implementation | Initializable, ContextUpgradeable,
IERC20Upgradeable, IERC20MetadataUpgradeable |||
| L | __ERC20_init | Internal | 🔒 | ● | initializer |
| L | __ERC20_init_unchained | Internal | 🔒 | ● | initializer |
| L | name | Public | ! | | NO ! |
| L | symbol | Public | ! | | NO ! |
| L | decimals | Public | ! | | NO ! |
| L | totalSupply | Public | ! | | NO ! |
| L | balanceOf | Public | ! | | NO ! |
| L | transfer | Public | ! | ● | NO ! |
| L | allowance | Public | ! | | NO ! |
| L | approve | Public | ! | ● | NO ! |
| L | transferFrom | Public | ! | ● | NO ! |
| L | increaseAllowance | Public | ! | ● | NO ! |
| L | decreaseAllowance | Public | ! | ● | NO ! |
| L | _transfer | Internal | 🔒 | ● | |
| L | _mint | Internal | 🔒 | ● | |
| L | _burn | Internal | 🔒 | ● | |
| L | _approve | Internal | 🔒 | ● | |
| L | _beforeTokenTransfer | Internal | 🔒 | ● | |
| L | _afterTokenTransfer | Internal | 🔒 | ● | |
|||||
| **OwnableUpgradeable** | Implementation | Initializable, ContextUpgradeable |||
| L | __Ownable_init | Internal | 🔒 | ● | initializer |
| L | __Ownable_init_unchained | Internal | 🔒 | ● | initializer |
| L | owner | Public | ! | | NO ! |
| L | renounceOwnership | Public | ! | ● | onlyOwner |
| L | transferOwnership | Public | ! | ● | onlyOwner |
| L | _setOwner | Private | 🔒 | ● | |
|||||
| **IUniswapV2Pair** | Interface | |||
| L | name | External | ! | | NO ! |
| L | symbol | External | ! | | NO ! |
| L | decimals | External | ! | | NO ! |
| L | totalSupply | External | ! | | NO ! |
| L | balanceOf | External | ! | | NO ! |
| L | allowance | External | ! | | NO ! |
| L | approve | External | ! | ● | NO ! |
| L | transfer | External | ! | ● | NO ! |
| L | transferFrom | External | ! | ● | NO ! |
| L | DOMAIN_SEPARATOR | External | ! | | NO ! |
| L | PERMIT_TYPEHASH | External | ! | | NO ! |
| L | nonces | External | ! | | NO ! |
| L | permit | External | ! | ● | NO ! |

```


FUNCTION DETAILS

```

| L | MINIMUM_LIQUIDITY | External ! | | NO ! |
| L | factory | External ! | | NO ! |
| L | token0 | External ! | | NO ! |
| L | token1 | External ! | | NO ! |
| L | getReserves | External ! | | NO ! |
| L | price0CumulativeLast | External ! | | NO ! |
| L | price1CumulativeLast | External ! | | NO ! |
| L | kLast | External ! | | NO ! |
| L | mint | External ! | ● | NO ! |
| L | burn | External ! | ● | NO ! |
| L | swap | External ! | ● | NO ! |
| L | skim | External ! | ● | NO ! |
| L | sync | External ! | ● | NO ! |
| L | initialize | External ! | ● | NO ! |
|||||
| **SafeMathInt** | Library | |||
| L | mul | Internal 🔒 | | |
| L | div | Internal 🔒 | | |
| L | sub | Internal 🔒 | | |
| L | add | Internal 🔒 | | |
| L | abs | Internal 🔒 | | |
| L | toUint256Safe | Internal 🔒 | | |
|||||
| **SafeMathUint** | Library | |||
| L | toInt256Safe | Internal 🔒 | | |
|||||
| **IterableMapping** | Library | |||
| L | get | Internal 🔒 | | |
| L | getIndexOfKey | Internal 🔒 | | |
| L | getKeyAtIndex | Internal 🔒 | | |
| L | size | Internal 🔒 | | |
| L | set | Internal 🔒 | ● | |
| L | remove | Internal 🔒 | ● | |
|||||
| **DividendPayingTokenInterface** | Interface | |||
| L | dividendOf | External ! | | NO ! |
| L | withdrawDividend | External ! | ● | NO ! |
|||||
| **DividendPayingTokenOptionalInterface** | Interface | |||
| L | withdrawableDividendOf | External ! | | NO ! |
| L | withdrawnDividendOf | External ! | | NO ! |
| L | accumulativeDividendOf | External ! | | NO ! |
|||||
| **DividendPayingToken** | Implementation | ERC20Upgradeable, OwnableUpgradeable,
DividendPayingTokenInterface, DividendPayingTokenOptionalInterface |||
| L | __DividendPayingToken_init | Internal 🔒 | ● | initializer |
| L | distributeCAKEDividends | Public ! | ● | onlyOwner |

```

FUNCTION DETAILS

```

| L | withdrawDividend | Public | ! | ● | NO | ! |
| L | _withdrawDividendOfUser | Internal | 🔒 | ● | ||
| L | dividendOf | Public | ! | | NO | ! |
| L | withdrawableDividendOf | Public | ! | | NO | ! |
| L | withdrawnDividendOf | Public | ! | | NO | ! |
| L | accumulativeDividendOf | Public | ! | | NO | ! |
| L | _transfer | Internal | 🔒 | ● | ||
| L | _mint | Internal | 🔒 | ● | ||
| L | _burn | Internal | 🔒 | ● | ||
| L | _setBalance | Internal | 🔒 | ● | ||
|||||
**BABYTOKENDividendTracker** | Implementation | OwnableUpgradeable,
DividendPayingToken |||
| L | initialize | External | ! | ● | initializer | |
| L | _transfer | Internal | 🔒 | ● | ||
| L | withdrawDividend | Public | ! | | NO | ! |
| L | excludeFromDividends | External | ! | ● | onlyOwner |
| L | isExcludedFromDividends | Public | ! | | NO | ! |
| L | updateClaimWait | External | ! | ● | onlyOwner |
| L | updateMinimumTokenBalanceForDividends | External | ! | ● | onlyOwner |
| L | getLastProcessedIndex | External | ! | | NO | ! |
| L | getNumberOfTokenHolders | External | ! | | NO | ! |
| L | getAccount | Public | ! | | NO | ! |
| L | getAccountAtIndex | Public | ! | | NO | ! |
| L | canAutoClaim | Private | 🔒 | ● | ||
| L | setBalance | External | ! | ● | onlyOwner |
| L | process | Public | ! | ● | NO | ! |
| L | processAccount | Public | ! | ● | onlyOwner |
|||||
**BaseToken** | Implementation | |||
|||||
**BABYTOKEN** | Implementation | ERC20, Ownable, BaseToken |||
| L | <Constructor> | Public | ! | 🟢 | ERC20 | |
| L | <Receive Ether> | External | ! | 🟢 | NO | ! |
| L | setSwapTokensAtAmount | External | ! | ● | onlyOwner |
| L | excludeFromFees | External | ! | ● | onlyOwner |
| L | excludeMultipleAccountsFromFees | External | ! | ● | onlyOwner |
| L | setMarketingWallet | External | ! | ● | onlyOwner |
| L | setTokenRewardsFee | External | ! | ● | onlyOwner |
| L | setLiquiditFee | External | ! | ● | onlyOwner |
| L | setMarketingFee | External | ! | ● | onlyOwner |
| L | _setAutomatedMarketMakerPair | Private | 🔒 | ● | ||
| L | updateGasForProcessing | Public | ! | ● | onlyOwner |
| L | updateClaimWait | External | ! | ● | onlyOwner |
| L | getClaimWait | External | ! | | NO | ! |
| L | updateMinimumTokenBalanceForDividends | External | ! | ● | onlyOwner |

```

FUNCTION DETAILS

```

| L | getMinimumTokenBalanceForDividends | External ! | [NO ! | |
| L | getTotalDividendsDistributed | External ! | [NO ! |
| L | isExcludedFromFees | Public ! | [NO ! |
| L | withdrawableDividendOf | Public ! | [NO ! |
| L | dividendTokenBalanceOf | Public ! | [NO ! |
| L | excludeFromDividends | External ! | ● | onlyOwner |
| L | isExcludedFromDividends | Public ! | [NO ! |
| L | getAccountDividendsInfo | External ! | [NO ! |
| L | getAccountDividendsInfoAtIndex | External ! | [NO ! |
| L | processDividendTracker | External ! | ● [NO ! |
| L | claim | External ! | ● [NO ! |
| L | getLastProcessedIndex | External ! | [NO ! |
| L | getNumberOfDividendTokenHolders | External ! | [NO ! |
| L | _transfer | Internal 🔒 | ● ||
| L | swapAndSendToFee | Private 🔒 | ● ||
| L | swapAndLiquify | Private 🔒 | ● ||
| L | swapTokensForEth | Private 🔒 | ● ||
| L | swapTokensForCake | Private 🔒 | ● ||
| L | addLiquidity | Private 🔒 | ● ||
| L | swapAndSendDividends | Private 🔒 | ● ||

```

Legend

```

| Symbol | Meaning |
|:-----:|:-----|
| ● | Function can modify state |
| 📄 | Function is payable |

```


TESTNET VERSION

Adding Liquidity 

Tx:

<https://testnet.bscscan.com/tx/0xf3f84c81f8a5f85164662ea9f036cc575e671541d41c55c0b6d837a7dcc20b0b>

=====

Buying from a fee excluded wallet 

Tx (0% tax):

<https://testnet.bscscan.com/tx/0x1a622b3df0a7534fc5dd531283c22d0eff04cbc499853afd8991a334e8070fd5>

=====

Selling from a fee excluded wallet 

Tx (0% tax):

<https://testnet.bscscan.com/tx/0x6df98d477e7982bc266ff80ae2edcd54d3965ce7d4b5364e72d92249a50ccdbc>

=====

Transferring using a fee excluded wallet 

Tx (0% tax):

<https://testnet.bscscan.com/tx/0x34986db3e7b0b953e11fd4502edc5a269c313334c944268b4bbcd04cb3a2d354>

=====

Buying from a regular wallet 

Tx (0-25% tax):

<https://testnet.bscscan.com/tx/0x406b1b1cfb6976f8ea06f125b36a34d679341b4035afbcb849fed63ce674d75>

TESTNET VERSION

Selling from a regular wallet ✓

Tx (0-25% tax):

<https://testnet.bscscan.com/tx/0x68eacb93ddf640863bdb3588f7f256c4cef8e977fe1e3c74705ca5a79d9df27f>

=====

Transferring a regular wallet ✓

Tx (0-25%):

<https://testnet.bscscan.com/tx/0x3f8b608660387c29fbc0e67cfde8060bfbebc449bb928c344da8f4f65c99485f>

=====

Internal swap (marketing BNB, Auto-liquidity, rewards distribution) ✓

Tx:

<https://testnet.bscscan.com/tx/0x68eacb93ddf640863bdb3588f7f256c4cef8e977fe1e3c74705ca5a79d9df27f>

MANUAL REVIEW

Severity Criteria

Expelee assesses the severity of disclosed vulnerabilities according to methodology based on OWASP standards.

Vulnerabilities are divided into three primary risk categories:

High

Medium

Low

High-level considerations for vulnerabilities span the following key areas when conducting assessments:

- Malicious input handling
- Escalation of privileges
- Arithmetic
- Gas use

Overall Risk Severity				
Impact	HIGH	Medium	High	Critical
	MEDIUM	Low	Medium	High
	LOW	Note	Low	Medium
		LOW	MEDIUM	HIGH
	Likelihood			

MEDIUM RISK FINDING

Excessive Fees

Category: **Centralization**

Status: Open

Severity: **Medium**

Overview:

Owner is able to set up to 25% tax for buy/sell/transfers separately.

```
function setTokenRewardsFee(uint256 value) external  
onlyOwner {  
    tokenRewardsFee = value;  
    totalFees =  
tokenRewardsFee.add(liquidityFee).add(marketingFee);  
    require(totalFees <= 25, "Total fee is over 25%");  
}
```

```
function setLiquiditFee(uint256 value) external  
onlyOwner {  
    liquidityFee = value;  
    totalFees =  
tokenRewardsFee.add(liquidityFee).add(marketingFee);  
    require(totalFees <= 25, "Total fee is over 25%");  
}
```

MEDIUM RISK FINDING

```
function setMarketingFee(uint256 value) external  
onlyOwner {  
    marketingFee = value;  
    totalFees =  
tokenRewardsFee.add(liquidityFee).add(marketingFee);  
    require(totalFees <= 25, "Total fee is over 25%");  
}
```

Suggestion:

Ensure that fees follow bellow scheme, this amounts are suggested by pinksale safu criteria:

0 <= buy fees <= 10%

0 <= sell fees <= 10%

0 <= transfer fees <= 10%

ABOUT EXPELEE

Expelee is a product-based aspirational Web3 start-up. Coping up with numerous solutions for blockchain security and constructing a Web3 ecosystem from deal making platform to developer hosting open platform, while also developing our own commercial and sustainable blockchain.

 www.expelee.com

 [expeleeofficial](https://twitter.com/expeleeofficial)

 [expelee](https://medium.com/expelee)

 [Expelee](https://t.me/Expelee)

 [expelee](https://in.linkedin.com/expelee)

 [expelee_official](https://www.instagram.com/expelee_official)

 [expelee-co](https://github.com/expelee-co)

expelee

Building the Futuristic **Blockchain Ecosystem**

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. Team provides no guarantess against the sale of team tokens or the removal of liquidity by the project audited in this document.

Always do your own research and project yourselves from being scammed. The Expelee 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 Expelee receive a payment to manipulate those results or change the awarding badge that we will be adding in our website. Alway 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 Expelee team disclaims any liability for the resulting losses.

The logo for Expelee, featuring the word "expelee" in a stylized font. The "ex" is in white, and "pelee" is in orange. The letters are bold and modern.

Building the Futuristic **Blockchain Ecosystem**