expelee

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

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/0xfabC5c382C3D20e
FE97Ed77Bec8f2e4613f5caac#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 acces 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.
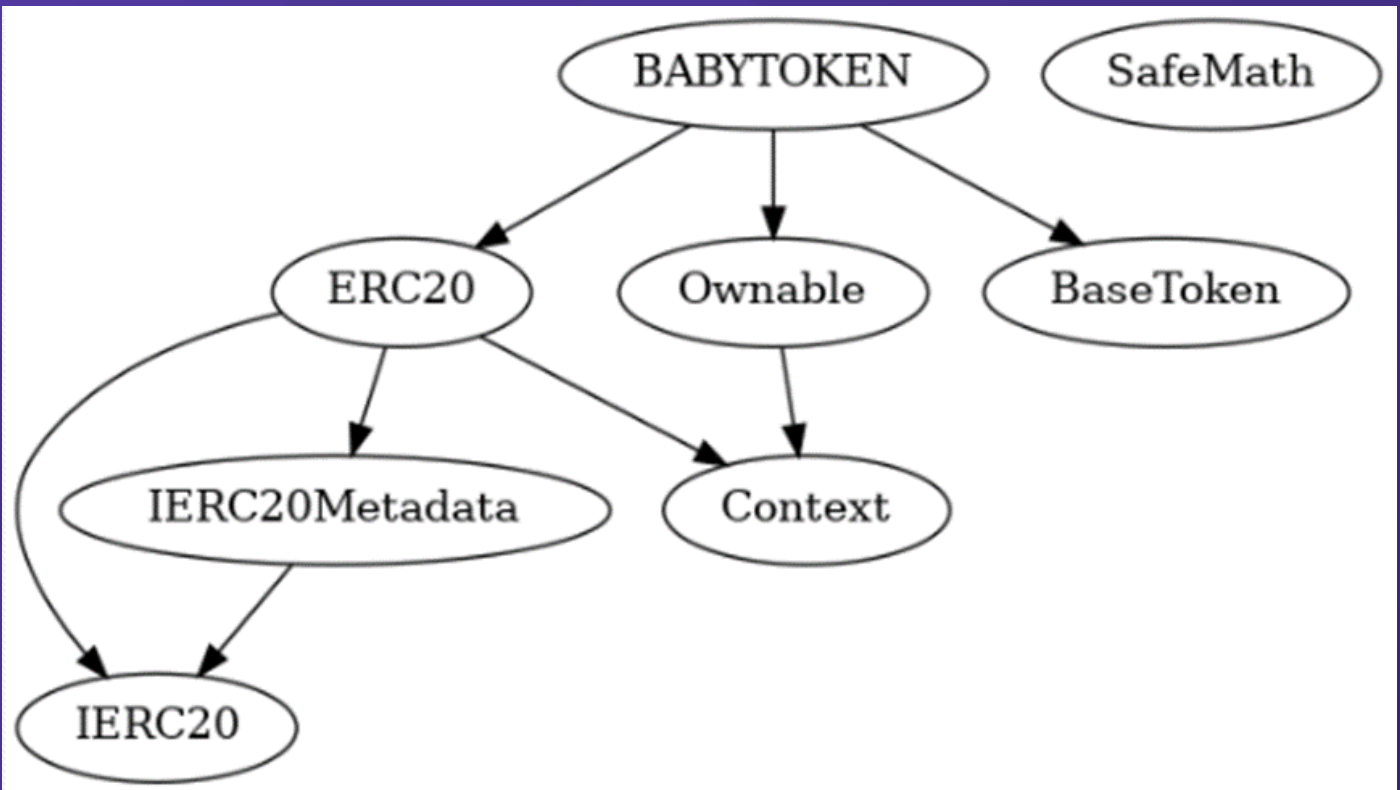
Low Risk

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

Informational

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

INHERITANCE TREES

FUNCTION DETAILS

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

└	factory	External	❗			NO ❗	
└	WETH	External	❗			NO ❗	
└	addLiquidity	External	❗		🔴	NO ❗	
└	addLiquidityETH	External	❗		💳	NO ❗	
└	removeLiquidity	External	❗		🔴	NO ❗	
└	removeLiquidityETH	External	❗		🔴	NO ❗	
└	removeLiquidityWithPermit	External	❗		🔴	NO ❗	
└	removeLiquidityETHWithPermit	External	❗		🔴	NO ❗	
└	swapExactTokensForTokens	External	❗		🔴	NO ❗	
└	swapTokensForExactTokens	External	❗		🔴	NO ❗	
└	swapExactETHForTokens	External	❗		💳	NO ❗	
└	swapTokensForExactETH	External	❗		🔴	NO ❗	
└	swapExactTokensForETH	External	❗		🔴	NO ❗	
└	swapETHForExactTokens	External	❗		💳	NO ❗	
└	quote	External	❗			NO ❗	
└	getAmountOut	External	❗			NO ❗	
└	getAmountIn	External	❗			NO ❗	
└	getAmountsOut	External	❗			NO ❗	
└	getAmountsIn	External	❗			NO ❗	
IUniswapV2Router02	Interface	IUniswapV2Router01		
└	removeLiquidityETHSupportingFeeOnTransferTokens	External	❗		🔴	NO ❗	
└	removeLiquidityETHWithPermitSupportingFeeOnTransferTokens	External	❗		🔴	NO ❗
└	swapExactTokensForTokensSupportingFeeOnTransferTokens	External	❗		🔴	NO ❗	
└	swapExactETHForTokensSupportingFeeOnTransferTokens	External	❗		💳	NO ❗	
└	swapExactTokensForETHSupportingFeeOnTransferTokens	External	❗		🔴	NO ❗	
IERC20Upgradeable	Interface			
└	totalSupply	External	❗			NO ❗	
└	balanceOf	External	❗			NO ❗	
└	transfer	External	❗		🔴	NO ❗	
└	allowance	External	❗			NO ❗	
└	approve	External	❗		🔴	NO ❗	
└	transferFrom	External	❗		🔴	NO ❗	
IERC20MetadataUpgradeable	Interface	IERC20Upgradeable		
└	name	External	❗			NO ❗	
└	symbol	External	❗			NO ❗	
└	decimals	External	❗			NO ❗	
Initializable	Implementation			
ContextUpgradeable	Implementation	Initializable		
└	__Context_init	Internal 🔒		🔴	initializer
└	__Context_init_unchained	Internal 🔒		🔴	initializer
└	_msgSender	Internal 🔒			

FUNCTION DETAILS

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

FUNCTION DETAILS

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

FUNCTION DETAILS

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

FUNCTION DETAILS

└	getMinimumTokenBalanceForDividends	External ❗		NO ❗
└	getTotalDividendsDistributed	External ❗		NO ❗
└	isExcludedFromFees	Public ❗		NO ❗
└	withdrawableDividendOf	Public ❗		NO ❗
└	dividendTokenBalanceOf	Public ❗		NO ❗
└	excludeFromDividends	External ❗	🔴	onlyOwner
└	isExcludedFromDividends	Public ❗		NO ❗
└	getAccountDividendsInfo	External ❗		NO ❗
└	getAccountDividendsInfoAtIndex	External ❗		NO ❗
└	processDividendTracker	External ❗	🔴	NO ❗
└	claim	External ❗	🔴	NO ❗
└	getLastProcessedIndex	External ❗		NO ❗
└	getNumberOfDividendTokenHolders	External ❗		NO ❗
└	_transfer	Internal 🔒	🔴	
└	swapAndSendToFee	Private 🔒	🔴	
└	swapAndLiquify	Private 🔒	🔴	
└	swapTokensForEth	Private 🔒	🔴	
└	swapTokensForCake	Private 🔒	🔴	
└	addLiquidity	Private 🔒	🔴	
└	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/0x34986db3e7b0b953e11fd4502edc5a269c313334c944268b4bbed04cb3a2d354

===

Buying from a regular wallet ✅
Tx (0-25% tax):
https://testnet.bscscan.com/tx/0x406b1b1cfb6976f8ea06f125b36a34d679341b4035afbacb849fed63ce674d75

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 standarts.

Vulnerabilities are dividend into three primary risk categroies:
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   Ⓜ expelee

✈ Expelee   in expelee

📷 expelee_official   ⓖ expelee-co

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.

Building the Futuristic Blockchain Ecosystem