

Smart Contract Security Audit V1

BabyTon Smart Contract Audit

May 11, 2024



<https://saferico.com/>

business@saferico.com

https://t.me/SFI_ANN

—

Table of Contents

Table of Contents

Background

Project Information

Token Smart Contract Information

Executive Summary

File and Function Level Report

File in Scope:

Issues Checking Status

SWC Attack Analysis

Severity Definitions

Audit Findings

Automatic testing

Testing proves

Inheritance graph

Call graph

Source lines

Risk level

Source units in scope

Capabilities

Unified Modeling Language (UML)

Functions signature

Automatic general report

Conclusion

Disclaimer

Background

The purpose of the audit was to achieve the following:

- Ensure that the smart contract functions as intended.
- Identify potential security issues with the smart contract.

The information in this report should be used to understand the risk exposure of the smart contract, and as a guide to improve the security posture of the smart contract by remediating the issues that were identified.

Project Information

- **Platform:** Binance Smart chain
- **Name:** BabyTon
- **Language :** Solidity
- **Contract Address:** 0xc5a092000c8dc5a1aceb88dd114c6f445c6e467a
- **Code Source:** <https://bscscan.com/token/0xc5a092000c8dc5a1aceb88dd114c6f445c6e467a#code>

Executive Summary

According to our assessment, the customer`s solidity smart contract is **Well-Secured**.

Well Secured	✓
Secured	
Poor Secured	
Insecure	

Automated checks are with remix IDE. All issues were performed by the team, which included the analysis of code functionality, manual audit found during automated analysis were manually reviewed and applicable vulnerabilities are presented in the audit overview section. The general overview is presented in the Project Information section and all issues found are located in the audit overview section.

Team found 0 critical, 0 high, 0 medium, 4 low, 0 very low-level issues and 0 note in all solidity files of the contract

The files:

BabyTon.sol

Audit Score:

99% secure



File and Function Level Report

File in Scope:

Contract Name	SHA 256 hash	Contract Address
BabyTon.sol	6810206e2c7363af128f47e1f743381c72a05f17	0xc5a092000c8dc5a1aceb88dd114c6f445c6e467a

- Contract: BabyTon
- Inherit: Context, IERC20, Ownable
- Observation: All passed including security check
- Test Report: passed
- Score: passed
- Conclusion: passed

Function	Test Result	Type / Return Type	Score
_developmentFee	✓	Read / public	Passed
_liquidityFee	✓	Read / public	Passed
balanceOf	✓	Read / public	Passed
allowance	✓	Read / public	Passed
_taxFee	✓	Read / public	Passed
decimals	✓	Read / public	Passed
reflectionFromToken	✓	Read / public	Passed
_maxTxAmount	✓	Read / public	Passed
blacklistMode	✓	Read / public	Passed
totalSupply	✓	Read / public	Passed
isBlacklisted	✓	Read / public	Passed
isExcludedFromFee	✓	Read / public	Passed
name	✓	Read / public	Passed
owner	✓	Read / public	Passed

isExcludedFromReward	✓	Read / public	Passed
swapAndLiquifyEnabled	✓	Read / public	Passed
totalFees	✓	Read / public	Passed
symbol	✓	Read / public	Passed
tokenFromReflection	✓	Read / public	Passed
uniswapV2Router	✓	Read / public	Passed
uniswapV2Pair	✓	Read / public	Passed
excludeFromReward	✓	Write / public	Passed
excludeFromFee	✓	Write / public	Passed
approve	✓	Write / public	Passed
enable_blacklist	✓	Write / public	Passed
deliver	✓	Write / public	Passed
decreaseAllowance	✓	Write / public	Passed
increaseAllowance	✓	Write / public	Passed
transferOwnership	✓	Write / public	Passed
renounceOwnership	✓	Write / public	Passed
includeInReward	✓	Write / public	Passed
includeInFee	✓	Write / public	Passed
manage_blacklist	✓	Write / public	Passed
transferFrom	✓	Write / public	Passed
transfer	✓	Write / public	Passed
setDevelopmentFeePercent	✓	Write / public	Passed
setLiquidityFeePercent	✓	Write / public	Passed
setMaxTxPercent	✓	Write / public	Passed
setSwapAndLiquifyEnabled	✓	Write / public	Passed
setTaxFeePercent	✓	Write / public	Passed

Issues Checking Status

SWC Attack Analysis

The Smart Contract Weakness Classification Registry (SWC Registry) is an implementation of the weakness classification scheme proposed in EIP-1470. It is loosely aligned to the terminologies and structure used in the Common Weakness Enumeration (CWE) for more info check

<https://swcregistry.io/>

No.	Issue Description	Checking Status
136	Unencrypted Private Data On-Chain	Passed
135	Code With No Effects	Passed
134	Message call with hardcoded gas amount	Passed
133	Hash Collisions With Multiple Variable Length Arguments	Passed
132	Unexpected Ether balance	Passed
131	Presence of unused variables	Passed
130	Right-To-Left-Override control character (U+202E)	Passed
129	Typographical Error	Passed
128	DoS with block gas limit.	Passed
127	Arbitrary Jump with Function Type Variable	Passed
126	Insufficient Gas Griefing	Passed
125	Incorrect Inheritance Order	Passed
124	Write to Arbitrary Storage Location	Passed
123	Requirement Violation	Passed
122	Lack of Proper Signature Verification	Passed
121	Missing Protection against Signature Replay Attacks	Passed
120	Weak Sources of Randomness from Chain Attributes	Passed
119	Shadowing State Variables	Passed

118	Incorrect Constructor Name	Passed
117	Signature Malleability	Passed
116	Block values as a proxy for time	Not Passed
115	Authorization through tx.origin	Passed
114	Transaction Order Dependence	Passed
113	DoS with Failed Call	Passed
112	Delegatecall to Untrusted Callee	Passed
111	Use of Deprecated Solidity Functions	Passed
110	Assert Violation	Passed
109	Uninitialized Storage Pointer	Passed
108	State Variable Default Visibility	Passed
107	Reentrancy	Passed
106	Unprotected SELFDESTRUCT Instruction	Passed
105	Unprotected Ether Withdrawal	Passed
104	Unchecked Call Return Value	Passed
103	Floating Pragma	Not Passed
102	Outdated Compiler Version	Passed
101	Integer Overflow and Underflow	Passed
100	Function Default Visibility	Passed

Severity Definitions

Risk Level	Description
Critical	Critical vulnerabilities are usually straightforward to exploit and can lead to tokens loss etc.
High	High-level vulnerabilities are difficult to exploit; however, they also have significant impact on smart contract execution, e.g. public access to crucial functions
Medium	Medium-level vulnerabilities are important to fix; however, they can't lead to tokens lose
Low	Low-level vulnerabilities are mostly related to outdated, unused etc. code snippets, that can't have significant impact on execution
Note	Lowest-level vulnerabilities, code style violations and info statements can't affect smart contract execution and can be ignored.

Audit Findings

Critical:

No Critical severity vulnerabilities were found.

High:

No High severity vulnerabilities were found.

Medium:

No Medium severity vulnerabilities were found.

Low:

#Owner privileges (In the period when the owner isn't renounced)

Description

The owner can change the Fees.

The owner can add any address to the blacklist.

The owner can include / exclude any address from reward or fees.

```
function manage_blacklist(address[] calldata addresses, bool status) public
onlyOwner {
    for (uint256 i; i < addresses.length; ++i) {
        isBlacklisted[addresses[i]] = status;
    }
}

function excludeFromFee(address account) public onlyOwner {
    _isExcludedFromFee[account] = true;
}

function includeInFee(address account) public onlyOwner {
    _isExcludedFromFee[account] = false;
}

function setTaxFeePercent(uint256 taxFee) external onlyOwner() {
    _taxFee = taxFee;
}

function setDevelopmentFeePercent(uint256 developmentFee) external onlyOwner()
{
    _developmentFee = developmentFee;
}

function setLiquidityFeePercent(uint256 liquidityFee) external onlyOwner() {
    _liquidityFee = liquidityFee;
}

function excludeFromReward(address account) public onlyOwner() {
    require(!_isExcluded[account], "Account is already excluded");
```

```

        if(_rOwned[account] > 0) {
            _tOwned[account] = tokenFromReflection(_rOwned[account]);
        }
        _isExcluded[account] = true;
        _excluded.push(account);
    }
    function includeInReward(address account) external onlyOwner() {
        require(!_isExcluded[account], "Account is already included");
        for (uint256 i = 0; i < _excluded.length; i++) {
            if (_excluded[i] == account) {
                _excluded[i] = _excluded[_excluded.length - 1];
                _tOwned[account] = 0;
                _isExcluded[account] = false;
                _excluded.pop();
                break;
            }
        }
    }
}

```

Remediation

Make these functions internal in next version or the team should announce the investors before doing anything to give them time if they want to do anything.

P.S: This issue is common to the majority of those smart contracts.

Status: **Acknowledged**.

#Pragam version not fixed

Description

It is a good practice to lock the solidity version for a live deployment (use 0.8.25 instead of ^0.8.4). contracts should be deployed with the same compiler version and flags that they have been tested the most with. Locking the pragma helps ensure that contracts do not accidentally get deployed using, for example, the latest compiler which may have higher risks of undiscovered bugs. Contracts may also be deployed by others and the pragma indicates the compiler version intended by the original authors. And avoid Solidity compiler Bugs check here

<https://sepolia.etherscan.io/solcbuginfo>

Remediation

Remove the ^ sign to lock the pragma version.

Status: **Acknowledged**.

#Missing zero address validation

When the owner wants add addresses to the blacklist, he has to check for the zero address to make. Otherwise, the function will not work fine.

```
function manage_blacklist(address[] calldata addresses, bool status) public
onlyOwner {
    for (uint256 i; i < addresses.length; ++i) {
        isBlacklisted[addresses[i]] = status;
    }
}
```

Remediation

Use the require statement to check for zero addresses.

Status: **Acknowledged.**

Use of block.timestamp for comparisons

The value of block.timestamp can be manipulated by the miner. And conditions with strict equality is difficult to achieve - block.timestamp.

```
function addLiquidity(uint256 tokenAmount, uint256 ethAmount)
private {
    _approve(address(this), address(uniswapV2Router),
tokenAmount);
    uniswapV2Router.addLiquidityETH{value: ethAmount}(
        address(this),
        tokenAmount,0,0,
        owner(),
        block.timestamp;)
```

Recommendation

Avoid use of block.timestamp.

Status

Acknowledged.

Very Low:

No Very Low severity vulnerabilities were found.

Notes:

No Notes were found.

Automatic Testing

1- SOLIDITY STATIC ANALYSIS

The image shows two side-by-side panels of the Solidity Static Analysis tool. Both panels have a title bar 'SOLIDITY STATIC ANALYSIS' and a 'Run' button. The left panel has checkboxes for 'Select all' and 'Autorun'. It contains two main sections: 'Security' and 'Gas & Economy'. The 'Security' section includes rules like 'Transaction origin', 'Check-effects-interaction', 'Inline assembly', 'Block timestamp', 'Low level calls', 'Block hash', and 'Selfdestruct'. The 'Gas & Economy' section includes rules like 'Gas costs', 'This on local calls', 'Delete dynamic array', 'For loop over dynamic array', and 'Ether transfer in loop'. The right panel has a section for 'ERC' with a rule for 'ERC20' and a section for 'Miscellaneous' with rules like 'Constant/View/Pure functions', 'Similar variable names', 'No return', 'Guard conditions', 'Result not used', 'String length', 'Delete from dynamic array', and 'Data truncated'.

SOLIDITY STATIC ANALYSIS

☒ Select all ☒ Autorun **Run**

Security

☒ Select Security

- ☒ **Transaction origin:**
'tx.origin' used
- ☒ **Check-effects-interaction:**
Potential reentrancy bugs
- ☒ **Inline assembly:**
Inline assembly used
- ☒ **Block timestamp:**
Can be influenced by miners
- ☒ **Low level calls:**
Should only be used by experienced devs
- ☒ **Block hash:**
Can be influenced by miners
- ☒ **Selfdestruct:**
Contracts using destructed contract can be broken

Gas & Economy

☒ Select Gas & Economy

- ☒ **Gas costs:**
Too high gas requirement of functions
- ☒ **This on local calls:**
Invocation of local functions via 'this'
- ☒ **Delete dynamic array:**
Use require/assert to ensure complete deletion
- ☒ **For loop over dynamic array:**
Iterations depend on dynamic array's size
- ☒ **Ether transfer in loop:**
Transferring Ether in a for/while/do-while loop

SOLIDITY STATIC ANALYSIS

ERC

☒ Select ERC

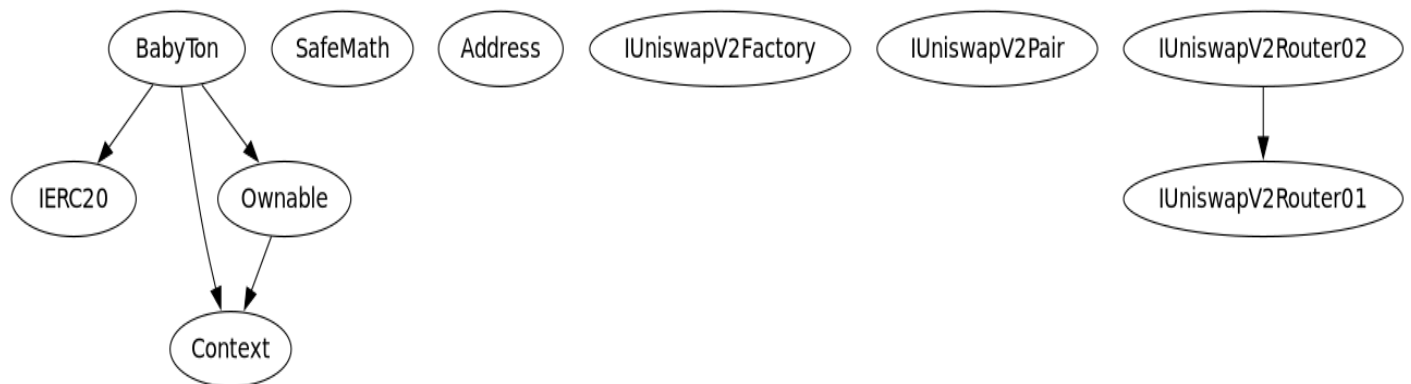
- ☒ **ERC20:**
'decimals' should be 'uint8'

Miscellaneous

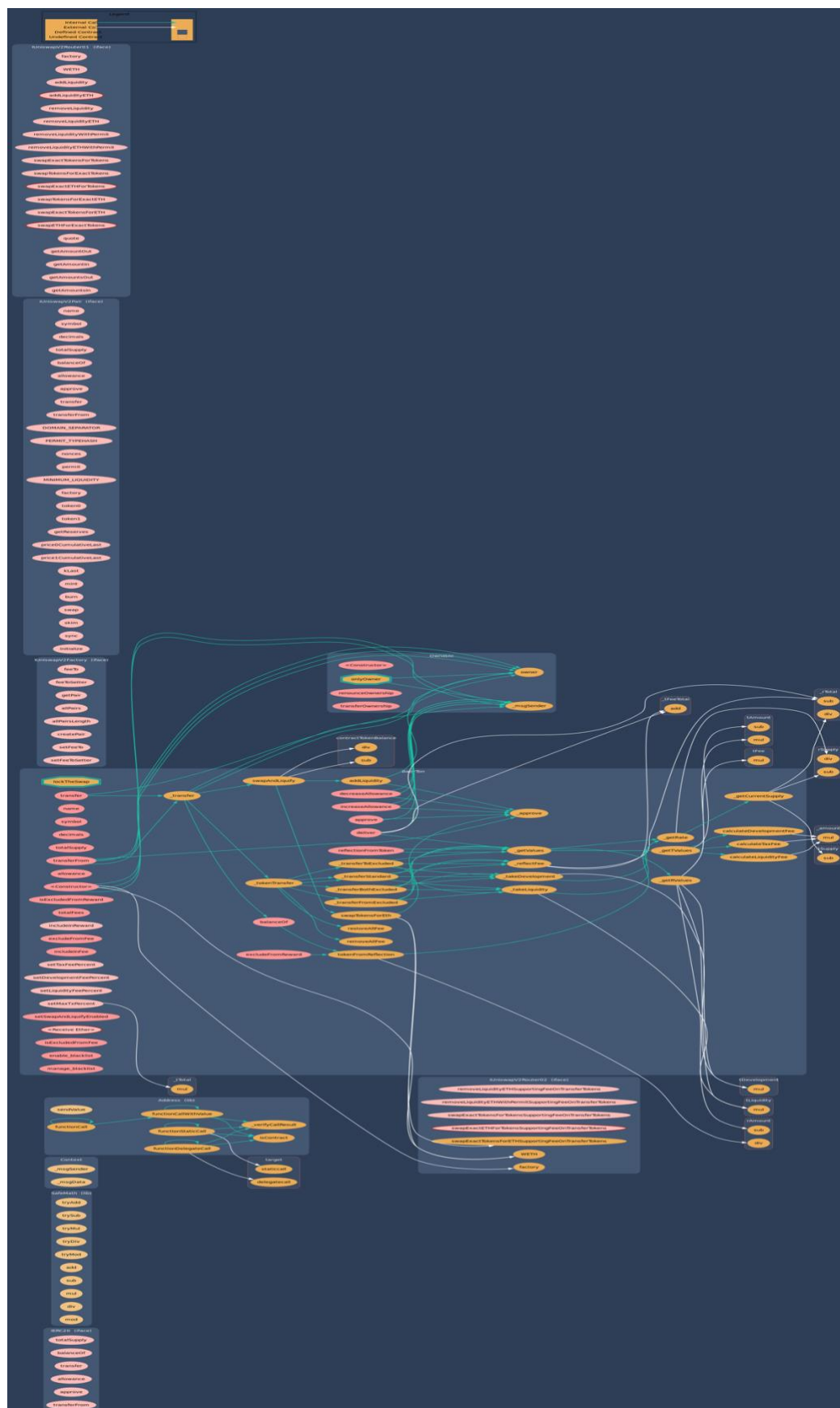
☒ Select Miscellaneous

- ☒ **Constant/View/Pure functions:**
Potentially constant/view/pure functions
- ☒ **Similar variable names:**
Variable names are too similar
- ☒ **No return:**
Function with 'returns' not returning
- ☒ **Guard conditions:**
Ensure appropriate use of require/assert
- ☒ **Result not used:**
The result of an operation not used
- ☒ **String length:**
Bytes length != String length
- ☒ **Delete from dynamic array:**
'delete' leaves a gap in array
- ☒ **Data truncated:**
Division on int/uint values truncates the result

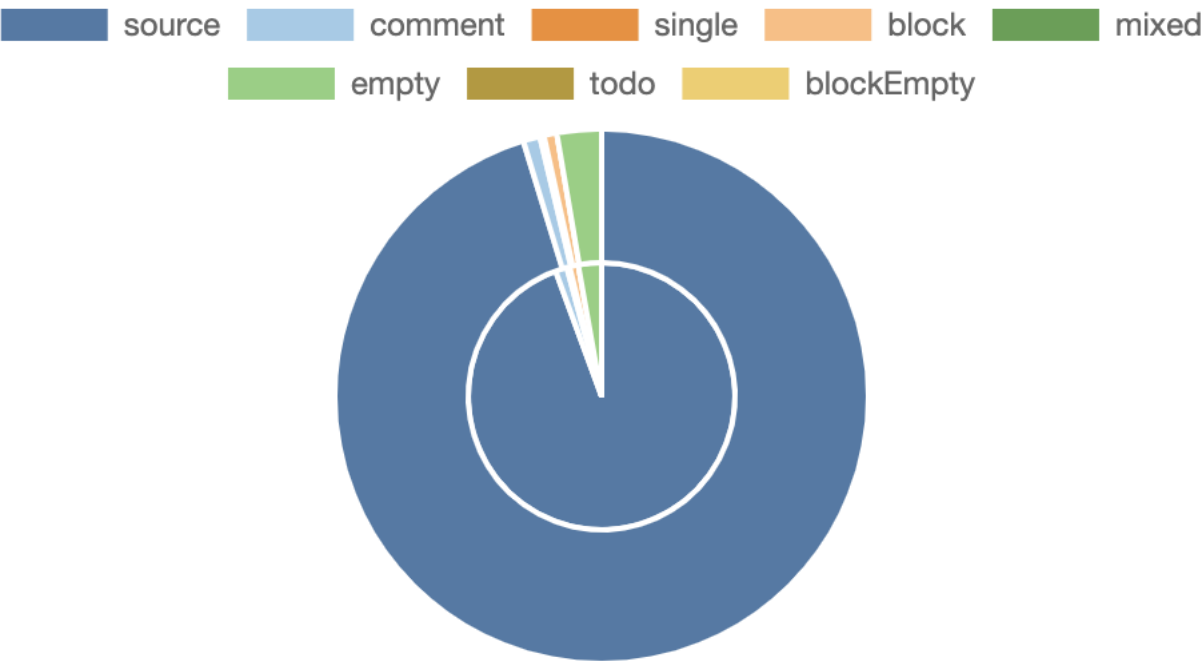
2- Inheritance graph



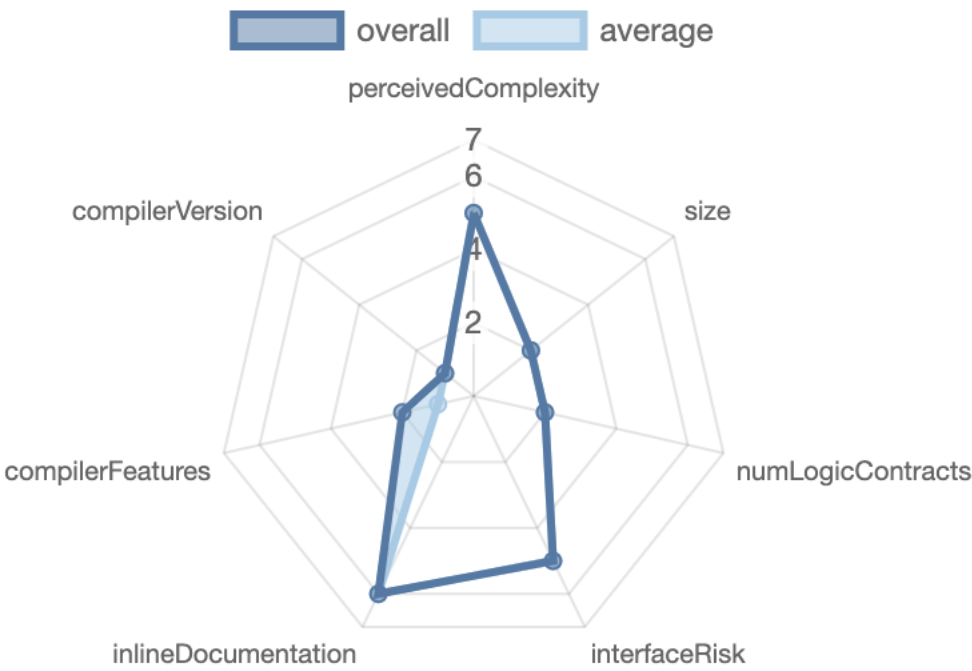
3- Call graph



Source lines







Risk level



Source units in scope

Source Units in Scope

Source Units Analyzed: 1
Source Units in Scope: 1 (100%)

Type	File	Logic Contracts	Interfaces	Lines	nLines	nSLOC	Comment Lines	Complex. Score	Capabilities
	BabyTon.sol	5	5	772	562	538	8	556	
	Totals	5	5	772	562	538	8	556	

Legend: [-]

- **Lines**: total lines of the source unit
- **nLines**: normalized lines of the source unit (e.g. normalizes functions spanning multiple lines)
- **nSLOC**: normalized source lines of code (only source-code lines; no comments, no blank lines)
- **Comment Lines**: lines containing single or block comments
- **Complexity Score**: a custom complexity score derived from code statements that are known to introduce code complexity (branches, loops, calls, external interfaces, ...)

Capabilities

Components

 Contracts	 Libraries	 Interfaces	 Abstract
1	2	5	2

Exposed Functions

This section lists functions that are explicitly declared public or payable. Please note that getter methods for public stateVars are not included.

 Public	 Payable
97	5

External	Internal	Private	Pure	View
71	97	24	25	46

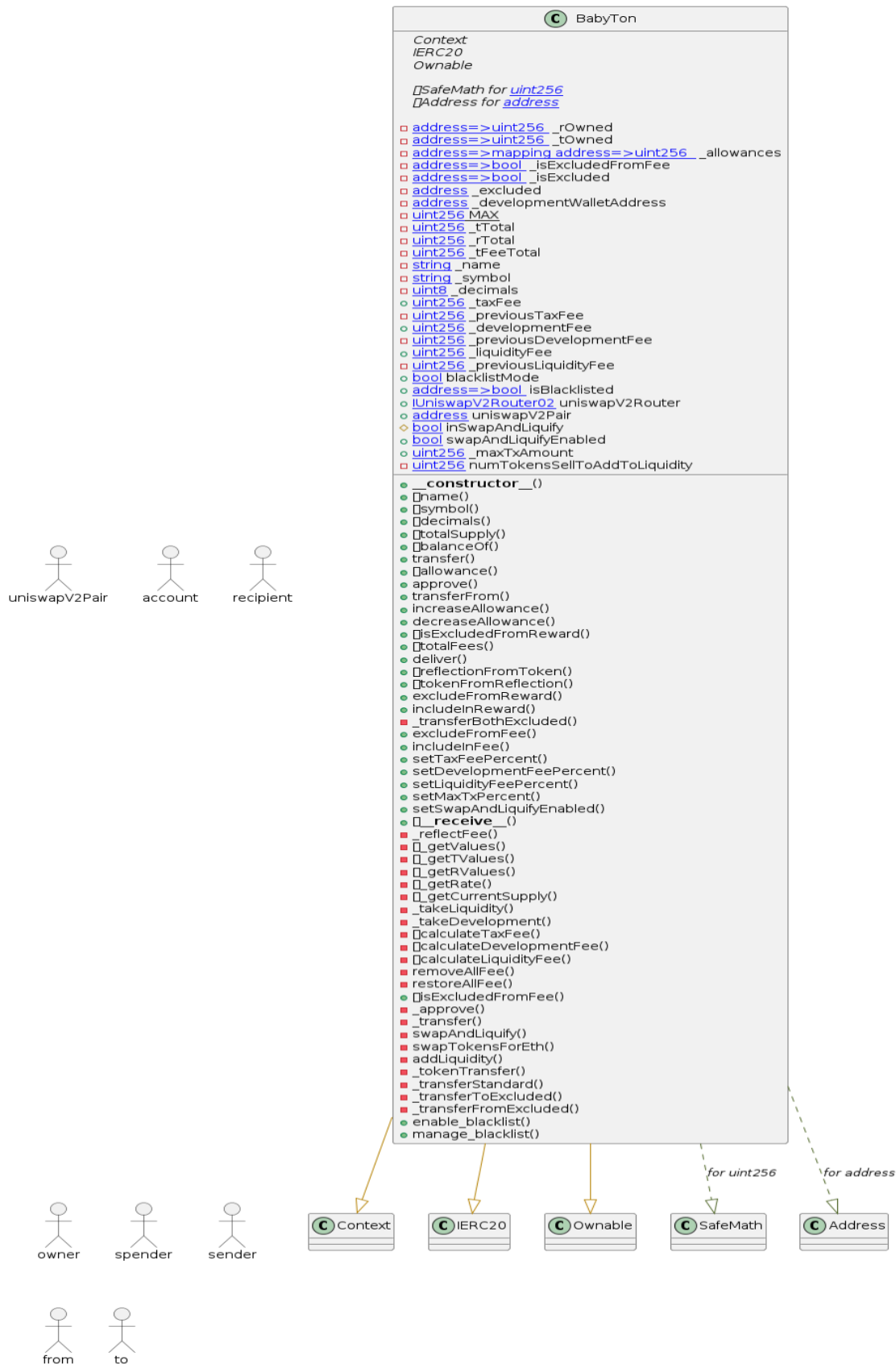
StateVariables

Total	 Public
29	9

Capabilities

Solidity Versions observed	🚧 Experimental Features	💰 Can Receive Funds	📺 Uses Assembly	🌐 Has Destroyable Contracts	
<input type="text" value="^0.8.4"/>		<input type="text" value="yes"/>	<input type="text" value="yes (2 asm blocks)"/>	<input type="text"/>	
👛 Transfers ETH	⚡ Low-Level Calls	👤 DelegateCall	🏠 Uses Hash Functions	📄 ECRECOVER	🌀 New/Create/Create2
<input type="text"/>	<input type="text"/>	<input type="text" value="yes"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Unified Modeling Language (UML)



Functions signature

Function Name	Sighash	Function Signature
totalSupply	18160ddd	totalSupply()
balanceOf	70a08231	balanceOf(address)
transfer	a9059cbb	transfer(address,uint256)
allowance	dd62ed3e	allowance(address,address)
approve	095ea7b3	approve(address,uint256)
transferFrom	23b872dd	transferFrom(address,address,uint256)
owner	8da5cb5b	owner()
renounceOwnership	715018a6	renounceOwnership()
transferOwnership	f2fde38b	transferOwnership(address)
feeTo	017e7e58	feeTo()
feeToSetter	094b7415	feeToSetter()
getPair	e6a43905	getPair(address,address)
allPairs	1e3dd18b	allPairs(uint256)
allPairsLength	574f2ba3	allPairsLength()
createPair	c9c65396	createPair(address,address)
setFeeTo	f46901ed	setFeeTo(address)
setFeeToSetter	a2e74af6	setFeeToSetter(address)
name	06fdde03	name()
symbol	95d89b41	symbol()
decimals	313ce567	decimals()
totalSupply	18160ddd	totalSupply()
balanceOf	70a08231	balanceOf(address)
allowance	dd62ed3e	allowance(address,address)
approve	095ea7b3	approve(address,uint256)
transfer	a9059cbb	transfer(address,uint256)
transferFrom	23b872dd	transferFrom(address,address,uint256)
DOMAIN_SEPARATOR	3644e515	DOMAIN_SEPARATOR()
PERMIT_TYPEHASH	30adf81f	PERMIT_TYPEHASH()
nonces	7ecebe00	nonces(address)
permit	d505accf	permit(address,address,uint256,uint256,uint8,bytes32,bytes32)
MINIMUM_LIQUIDITY	ba9a7a56	MINIMUM_LIQUIDITY()
factory	c45a0155	factory()
token0	0dfe1681	token0()
token1	d21220a7	token1()
getReserves	0902f1ac	getReserves()
price0CumulativeLast	5909c0d5	price0CumulativeLast()
price1CumulativeLast	5a3d5493	price1CumulativeLast()
kLast	7464fc3d	kLast()
mint	6a627842	mint(address)
burn	89afcb44	burn(address)
swap	022c0d9f	swap(uint256,uint256,address,bytes)
skim	bc25cf77	skim(address)
sync	fff6cae9	sync()
initialize	c4d66de8	initialize(address)
factory	c45a0155	factory()
WETH	ad5c4648	WETH()

```

| addLiquidity | e8e33700 |
addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uint
256) |
| addLiquidityETH | f305d719 |
addLiquidityETH(address,uint256,uint256,uint256,address,uint256) |
| removeLiquidity | baa2abde |
removeLiquidity(address,address,uint256,uint256,uint256,address,uint256)
|
| removeLiquidityETH | 02751cec |
removeLiquidityETH(address,uint256,uint256,uint256,address,uint256) |
| removeLiquidityWithPermit | 2195995c |
removeLiquidityWithPermit(address,address,uint256,uint256,uint256,address
,uint256,bool,uint8,bytes32,bytes32) |
| removeLiquidityETHWithPermit | ded9382a |
removeLiquidityETHWithPermit(address,uint256,uint256,uint256,address,uint
256,bool,uint8,bytes32,bytes32) |
| swapExactTokensForTokens | 38ed1739 |
swapExactTokensForTokens(uint256,uint256,address[],address,uint256) |
| swapTokensForExactTokens | 8803dbee |
swapTokensForExactTokens(uint256,uint256,address[],address,uint256) |
| swapExactETHForTokens | 7ff36ab5 |
swapExactETHForTokens(uint256,address[],address,uint256) |
| swapTokensForExactETH | 4a25d94a |
swapTokensForExactETH(uint256,uint256,address[],address,uint256) |
| swapExactTokensForETH | 18cbafe5 |
swapExactTokensForETH(uint256,uint256,address[],address,uint256) |
| swapETHForExactTokens | fb3bdb41 |
swapETHForExactTokens(uint256,address[],address,uint256) |
| quote | ad615dec | quote(uint256,uint256,uint256) |
| getAmountOut | 054d50d4 | getAmountOut(uint256,uint256,uint256) |
| getAmountIn | 85f8c259 | getAmountIn(uint256,uint256,uint256) |
| getAmountsOut | d06ca61f | getAmountsOut(uint256,address[]) |
| getAmountsIn | 1f00ca74 | getAmountsIn(uint256,address[]) |
| removeLiquidityETHSupportingFeeOnTransferTokens | af2979eb |
removeLiquidityETHSupportingFeeOnTransferTokens(address,uint256,uint256,u
int256,address,uint256) |
| removeLiquidityETHWithPermitSupportingFeeOnTransferTokens | 5b0d5984 |
removeLiquidityETHWithPermitSupportingFeeOnTransferTokens(address,uint256
,uint256,uint256,address,uint256,bool,uint8,bytes32,bytes32) |
| swapExactTokensForTokensSupportingFeeOnTransferTokens | 5c11d795 |
swapExactTokensForTokensSupportingFeeOnTransferTokens(uint256,uint256,add
ress[],address,uint256) |
| swapExactETHForTokensSupportingFeeOnTransferTokens | b6f9de95 |
swapExactETHForTokensSupportingFeeOnTransferTokens(uint256,address[],addr
ess,uint256) |
| swapExactTokensForETHSupportingFeeOnTransferTokens | 791ac947 |
swapExactTokensForETHSupportingFeeOnTransferTokens(uint256,uint256,addres
s[],address,uint256) |
| name | 06fdde03 | name() |
| symbol | 95d89b41 | symbol() |
| decimals | 313ce567 | decimals() |
| totalSupply | 18160ddd | totalSupply() |

```

```
| balanceOf | 70a08231 | balanceOf(address) |
| transfer | a9059cbb | transfer(address,uint256) |
| allowance | dd62ed3e | allowance(address,address) |
| approve | 095ea7b3 | approve(address,uint256) |
| transferFrom | 23b872dd | transferFrom(address,address,uint256) |
| increaseAllowance | 39509351 | increaseAllowance(address,uint256) |
| decreaseAllowance | a457c2d7 | decreaseAllowance(address,uint256) |
| isExcludedFromReward | 88f82020 | isExcludedFromReward(address) |
| totalFees | 13114a9d | totalFees() |
| deliver | 3bd5d173 | deliver(uint256) |
| reflectionFromToken | 4549b039 | reflectionFromToken(uint256,bool) |
| tokenFromReflection | 2d838119 | tokenFromReflection(uint256) |
| excludeFromReward | 52390c02 | excludeFromReward(address) |
| includeInReward | 3685d419 | includeInReward(address) |
| excludeFromFee | 437823ec | excludeFromFee(address) |
| includeInFee | ea2f0b37 | includeInFee(address) |
| setTaxFeePercent | 061c82d0 | setTaxFeePercent(uint256) |
| setDevelopmentFeePercent | 4680ff35 | setDevelopmentFeePercent(uint256) |
|
| setLiquidityFeePercent | 8ee88c53 | setLiquidityFeePercent(uint256) |
| setMaxTxPercent | d543dbeb | setMaxTxPercent(uint256) |
| setSwapAndLiquifyEnabled | c49b9a80 | setSwapAndLiquifyEnabled(bool) |
| isExcludedFromFee | 5342acb4 | isExcludedFromFee(address) |
| enable_blacklist | 5e562f3b | enable_blacklist(bool) |
| manage_blacklist | 8e2eee84 | manage_blacklist(address[],bool) |
```

Automatic general report

Files Description Table

File Name	SHA-1 Hash
/Users/macbook/Desktop/smart contracts/BabyTon.sol	6810206e2c7363af128f47e1f743381c72a05f17

Contracts Description Table

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!
SafeMath	Library		
L tryAdd	Internal	🔒	
L trySub	Internal	🔒	
L tryMul	Internal	🔒	
L tryDiv	Internal	🔒	
L tryMod	Internal	🔒	
L add	Internal	🔒	
L sub	Internal	🔒	
L mul	Internal	🔒	
L div	Internal	🔒	
L mod	Internal	🔒	
L sub	Internal	🔒	
L div	Internal	🔒	
L mod	Internal	🔒	
Context	Implementation		
L _msgSender	Internal	🔒	
L _msgData	Internal	🔒	
Address	Library		
L isContract	Internal	🔒	
L sendValue	Internal	🔒	
L functionCall	Internal	🔒	
L functionCall	Internal	🔒	

L	functionCallWithValue	Internal				
L	functionCallWithValue	Internal				
L	functionStaticCall	Internal				
L	functionStaticCall	Internal				
L	functionDelegateCall	Internal				
L	functionDelegateCall	Internal				
L	_verifyCallResult	Private				
Ownable Implementation Context						
L	<Constructor>	Public	!		NO	!
L	owner	Public	!		NO	!
L	renounceOwnership	Public	!			onlyOwner
L	transferOwnership	Public	!			onlyOwner
IUniswapV2Factory Interface						
L	feeTo	External	!		NO	!
L	feeToSetter	External	!		NO	!
L	getPair	External	!		NO	!
L	allPairs	External	!		NO	!
L	allPairsLength	External	!		NO	!
L	createPair	External	!		NO	!
L	setFeeTo	External	!		NO	!
L	setFeeToSetter	External	!		NO	!
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	!
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! |
| | | |
| **IUniswapV2Router01** | Interface | | |
| 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 | removeLiquidityETHWithPermit | 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! |
| | | | | |
| **BabyTon** | Implementation | Context, IERC20, Ownable | |
| 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 | isExcludedFromReward | Public ! | NO! |
| L | totalFees | Public ! | NO! |
| L | deliver | Public ! | ⬤ | NO! |

```

L	reflectionFromToken	Public	!		NO!	
L	tokenFromReflection	Public	!		NO!	
L	excludeFromReward	Public	!	⬛	onlyOwner	
L	includeInReward	External	!	⬛	onlyOwner	
L	_transferBothExcluded	Private	🔒	⬛		
L	excludeFromFee	Public	!	⬛	onlyOwner	
L	includeInFee	Public	!	⬛	onlyOwner	
L	setTaxFeePercent	External	!	⬛	onlyOwner	
L	setDevelopmentFeePercent	External	!	⬛	onlyOwner	
L	setLiquidityFeePercent	External	!	⬛	onlyOwner	
L	setMaxTxPercent	External	!	⬛	onlyOwner	
L	setSwapAndLiquifyEnabled	Public	!	⬛	onlyOwner	
L	<Receive Ether>	External	!	👤	NO!	
L	_reflectFee	Private	🔒	⬛		
L	_getValues	Private	🔒			
L	_getTValues	Private	🔒			
L	_getRValues	Private	🔒			
L	_getRate	Private	🔒			
L	_getCurrentSupply	Private	🔒			
L	_takeLiquidity	Private	🔒	⬛		
L	_takeDevelopment	Private	🔒	⬛		
L	calculateTaxFee	Private	🔒			
L	calculateDevelopmentFee	Private	🔒			
L	calculateLiquidityFee	Private	🔒			
L	removeAllFee	Private	🔒	⬛		
L	restoreAllFee	Private	🔒	⬛		
L	isExcludedFromFee	Public	!		NO!	
L	_approve	Private	🔒	⬛		
L	_transfer	Private	🔒	⬛		
L	swapAndLiquify	Private	🔒	⬛	lockTheSwap	
L	swapTokensForEth	Private	🔒	⬛		
L	addLiquidity	Private	🔒	⬛		
L	_tokenTransfer	Private	🔒	⬛		
L	_transferStandard	Private	🔒	⬛		
L	_transferToExcluded	Private	🔒	⬛		
L	_transferFromExcluded	Private	🔒	⬛		
L	enable_blacklist	Public	!	⬛	onlyOwner	
L	manage_blacklist	Public	!	⬛	onlyOwner	

Legend

Symbol	Meaning
:-----:	-----
⬛	Function can modify state
👤	Function is payable

Conclusion

The contracts are written systematically. Team found no critical issues. So, it is good to go for production.

Since possible test cases can be unlimited and developer level documentation (code flow diagram with function level description) not provided, for such an extensive smart contract protocol, we provide no such guarantee of future outcomes. We have used all the latest static tools and manual observations to cover maximum possible test cases to scan Everything.

Security state of the reviewed contract is “Well Secured”.

- ✓ No volatile code.
- ✓ No high severity issues were found.

Disclaimer

This is a limited report on our findings based on our analysis, in accordance with good industry practice as of 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 the team on the basis of what it says or doesn't say, or how team produced it, and it is important for you to conduct your own independent investigations before making any decisions. team go into more detail on this in the below disclaimer below – please make sure to read it in full.

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 Saferico and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (Saferico s) owe no duty of care towards you or any other person, nor does Saferico 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 Saferico 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, Saferico hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against Saferico, 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.