auditace.tech

AuditAce

FROM INCEPTION TO SUCCESS

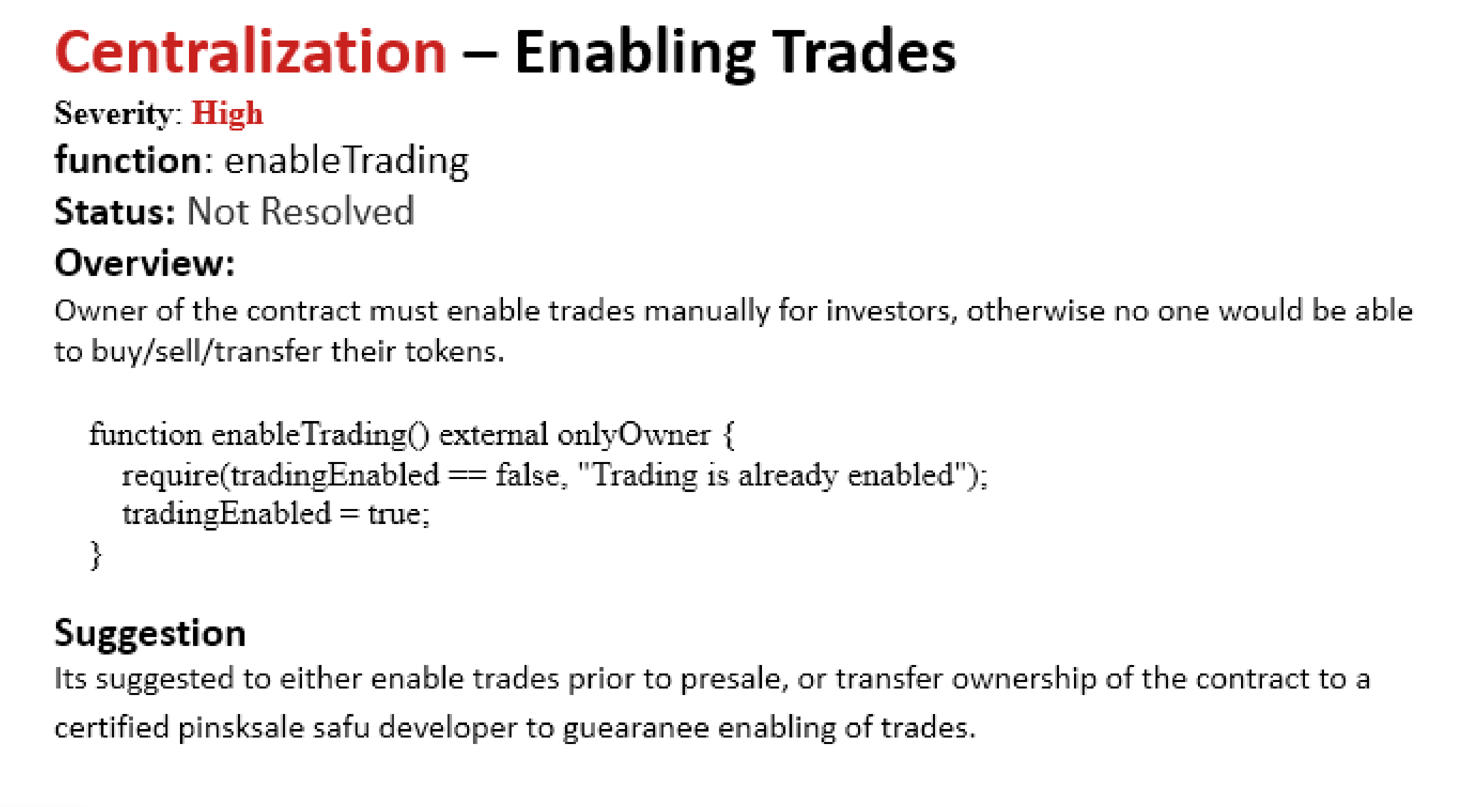
Smart Contract Audit

FOR

$Bcow 2.0

DATED : 3 July 23'

HIGH RISK FINDING



Centralization – Enabling Trades

Severity: High

function: enableTrading

Status: Not Resolved

Overview:

Owner of the contract must enable trades manually for investors, otherwise no one would be able to buy/sell/transfer their tokens.

```
function enableTrading() external onlyOwner {
    require(tradingEnabled == false, "Trading is already enabled");
    tradingEnabled = true;
}
```

Suggestion

Its suggested to either enable trades prior to presale, or transfer ownership of the contract to a certified pinsksale safu developer to guearanee enabling of trades.

AUDIT SUMMARY

Project name – $Bcow 2.0

Date: 3 July, 2023

Scope of Audit- Audit Ace was consulted to conduct the smart contract audit of the solidity source codes.

Audit Status: **Passed with High Risk**

Issues Found

Status	Critical	High	Medium	Low	Suggestion
Open	0	1	0	0	0
Acknowledged	0	0	0	0	0
Resolved	0	0	0	0	0

USED TOOLS

Tools:

1- Manual Review:
A line by line code review has been performed by audit ace team.

2- BSC Test Network: All tests were conducted on the BSC Test network, and each test has a corresponding transaction attached to it. These tests can be found in the "Functional Tests" section of the report.

3- Slither :
The code has undergone static analysis using Slither.

Testnet version:
The tests were performed using the contract deployed on the BSC Testnet, which can be found at the following address:
https://testnet.bscscan.com/token/0x6996B024A5978B31870b860A3e951F3f8074373c

Token Information

Token Name : Babycow 2.0

Token Symbol: $Bcow 2.0

Decimals: 9

Token Supply: 1,000,000,000

Token Address:
0x969C07a8ceD068EBc1926DB60157e0cfa1A81F38

Checksum:
eb3e8e8c1bc4f5dea1799b25ae684fed19e124cc

Owner:
0xb6B33f404Cd3eE241809bA2427F880d673b897D6
(at time of writing the audit)

Deployer:
0xb6B33f404Cd3eE241809bA2427F880d673b897D6

TOKEN OVERVIEW

Fees:

Buy Fees: 0%

Sell Fees: 5%

Transfer Fees: 0%

Fees Privilege: No fees

Ownership: Owned

Minting: none

Max Tx Amount/ Max Wallet Amount: No

Blacklist: No

Other Privileges: - Initial distribution of the tokens

- enabling trades

AUDIT METHODOLOGY

The auditing process will follow a routine as special considerations by Auditace:

- Review of the specifications, sources, and instructions provided to Auditace to make sure the contract logic meets the intentions of the client without exposing the user's funds to risk.

- Manual review of the entire codebase by our experts, which is the process of reading source code line-by-line in an attempt to identify potential vulnerabilities.

- Specification comparison is the process of checking whether the code does what the specifications, sources, and instructions provided to Auditace describe.

- Test coverage analysis determines whether the test cases are covering the code and how much code isexercised when we run the test cases.

- Symbolic execution is analysing a program to determine what inputs cause each part of a program to execute.

- Reviewing the codebase to improve maintainability, security, and control based on the established industry and academic practices.

VULNERABILITY CHECKLIST

- ✓ Return values of low-level calls
- ✓ Private modifier
- ✓ Multiple Sends
- ✓ Using Suicide
- ✓ Gas Limitand Loops
- ✓ Address hardcoded
- ✓ Exception Disorder
- ✓ Using inline assembly
- ✓ Divide before multiply
- ✓ Missing Zero Address Validation
- ✓ Compiler version not fixed
- ✓ **Gasless Send**
- ✓ Using block.timestamp
- ✓ Re-entrancy
- ✓ Tautology or contradiction
- ✓ Timestamp Dependence
- ✓ Revert/require functions
- ✓ Use of tx.origin
- ✓ Integer overflow/underflow
- ✓ Dangerous strict equalities
- ✓ Using SHA3
- ✓ Using throw

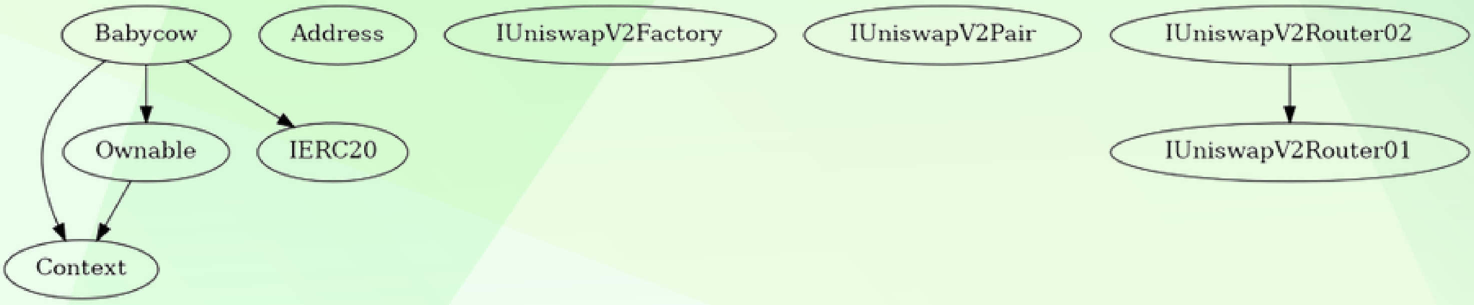
CLASSIFICATION OF RISK

Severity	Description
◆ **Critical**	These vulnerabilities could be exploited easily and can lead to asset loss, data loss, asset, or data manipulation. They should be fixed right away.
◆ **High-Risk**	A vulnerability that affects the desired outcome when using a contract, or provides the opportunity to use a contract in an unintended way.
◆ **Medium-Risk**	A vulnerability that could affect the desired outcome of executing the contract in a specific scenario.
◆ **Low-Risk**	A vulnerability that does not have a significant impact on possible scenarios for the use of the contract and is probably subjective.
◆ **Gas Optimization /Suggestion**	A vulnerability that has an informational character but is not affecting any of the code.

Findings

Severity	Found
◆ **Critical**	0
◆ **High-Risk**	1
◆ **Medium-Risk**	0
◆ **Low-Risk**	0
◆ **Gas Optimization / Suggestions**	0

INHERITANCE TREE



POINTS TO NOTE

- Owner is not change current fees (0% buy / 5% sell / 0% transfer)
- Owner is not able to blacklist an address
- Owner is not able to disable buy/sell/transfers
- Owner is not able to set max wallet limit and minimum wallet limits
- Owner is not able to mint new tokens
- Owner must enable trades manually

CONTRACT ASSESMENT

```
| Contract |        Type        |       Bases      |                  |                 |
|:----------:|:-------------------:|:----------------:|:----------------:|:---------------:|
|     └      |  **Function Name**  |  **Visibility**  |  **Mutability**  |  **Modifiers**  |
||||||
| **Context** | Implementation |  |||
| └ | _msgSender | Internal 🔒 |   | |
| └ | _msgData | Internal 🔒 |   | |
||||||
| **Ownable** | Implementation | Context |||
| └ | <Constructor> | Public ❗️ | 🛑  |NO ❗️ |
| └ | owner | Public ❗️ |   |NO ❗️ |
| └ | renounceOwnership | Public ❗️ | 🛑  | onlyOwner |
| └ | transferOwnership | Public ❗️ | 🛑  | onlyOwner |
||||||
| **IERC20** | Interface |  |||
| └ | totalSupply | External ❗️ |   |NO ❗️ |
| └ | balanceOf | External ❗️ |   |NO ❗️ |
| └ | transfer | External ❗️ | 🛑  |NO ❗️ |
| └ | allowance | External ❗️ |   |NO ❗️ |
| └ | approve | External ❗️ | 🛑  |NO ❗️ |
| └ | transferFrom | External ❗️ | 🛑  |NO ❗️ |
||||||
| **Address** | Library |  |||
| └ | isContract | Internal 🔒 |   | |
| └ | sendValue | Internal 🔒 | 🛑  | |
| └ | functionCall | Internal 🔒 | 🛑  | |
| └ | functionCall | Internal 🔒 | 🛑  | |
| └ | functionCallWithValue | Internal 🔒 | 🛑  | |
| └ | functionCallWithValue | Internal 🔒 | 🛑  | |
| └ | _functionCallWithValue | Private 🔐 | 🛑  | |
||||||
| **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 ❗️ |
||||||
| **IUniswapV2Pair** | Interface |  |||
| └ | name | External ❗️ |   |NO ❗️ |
| └ | symbol | External ❗️ |   |NO ❗️ |
```

CONTRACT ASSESMENT

```
| └ | 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 ❗ |
| └ | 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 ❗ |
| └ | burn | External ❗ | 🛑 |NO ❗ |
| └ | swap | External ❗ | 🛑 |NO ❗ |
| └ | skim | External ❗ | 🛑 |NO ❗ |
| └ | sync | External ❗ | 🛑 |NO ❗ |
| └ | initialize | External ❗ | 🛑 |NO ❗ |
||||||
| **IUniswapV2Router01** | Interface | |||
| └ | 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 ❗ |
```

CONTRACT ASSESMENT

```
| └ | 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 ❗ |
||||||
| **Babycow** | Implementation | Context, IERC20, Ownable |||
| └ | <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 ❗ |
| └ | isExcludedFromReward | Public ❗ | |NO ❗ |
| └ | totalReflectionDistributed | Public ❗ | |NO ❗ |
| └ | deliver | Public ❗ | 🛑 |NO ❗ |
| └ | reflectionFromToken | Public ❗ | |NO ❗ |
| └ | tokenFromReflection | Public ❗ | |NO ❗ |
| └ | excludeFromReward | Public ❗ | 🛑 | onlyOwner |
| └ | includeInReward | External ❗ | 🛑 | onlyOwner |
| └ | <Receive Ether> | External ❗ | 💵 |NO ❗ |
| └ | claimStuckTokens | External ❗ | 🛑 | onlyOwner |
| └ | _reflectFee | Private 🔐 | 🛑 | |
| └ | _getValues | Private 🔐 | | |
| └ | _getTValues | Private 🔐 | | |
| └ | _getRValues | Private 🔐 | | |
| └ | _getRate | Private 🔐 | | |
| └ | _getCurrentSupply | Private 🔐 | | |
| └ | _takeLiquidity | Private 🔐 | 🛑 | |
| └ | _takeMarketing | Private 🔐 | 🛑 | |
| └ | calculateTaxFee | Private 🔐 | | |
```

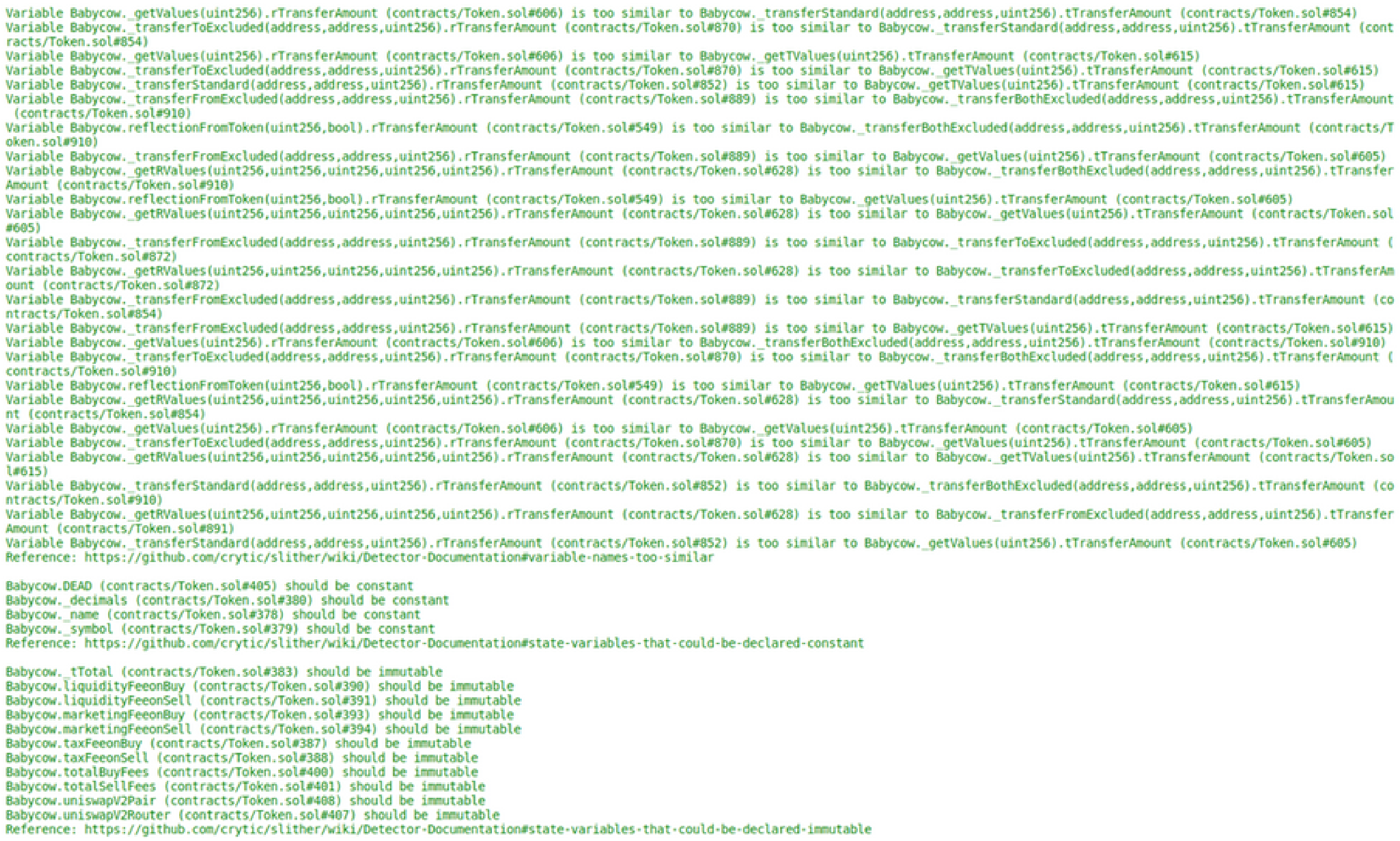
CONTRACT ASSESMENT

```
| └ | calculateLiquidityFee | Private 🔐 |   | |
| └ | calculateMarketingFee | Private 🔐 |   | |
| └ | removeAllFee | Private 🔐 | 🛑  | |
| └ | setBuyFee | Private 🔐 | 🛑  | |
| └ | setSellFee | Private 🔐 | 🛑  | |
| └ | isExcludedFromFee | Public ❗️ |   |NO❗️ |
| └ | _approve | Private 🔐 | 🛑  | |
| └ | enableTrading | External ❗️ | 🛑  | onlyOwner |
| └ | _transfer | Private 🔐 | 🛑  | |
| └ | swapAndLiquify | Private 🔐 | 🛑  | |
| └ | swapAndSendMarketing | Private 🔐 | 🛑  | |
| └ | setSwapTokensAtAmount | External ❗️ | 🛑  | onlyOwner |
| └ | setSwapEnabled | External ❗️ | 🛑  | onlyOwner |
| └ | _tokenTransfer | Private 🔐 | 🛑  | |
| └ | _transferStandard | Private 🔐 | 🛑  | |
| └ | _transferToExcluded | Private 🔐 | 🛑  | |
| └ | _transferFromExcluded | Private 🔐 | 🛑  | |
| └ | _transferBothExcluded | Private 🔐 | 🛑  | |
| └ | excludeFromFees | External ❗️ | 🛑  | onlyOwner |
| └ | changeMarketingWallet | External ❗️ | 🛑  | onlyOwner |


### Legend

|  Symbol  |  Meaning  |
|:--------:|-----------|
|    🛑    | Function can modify state |
|    💵    | Function is payable |
```

STATIC ANALYSIS



```
Variable Babycow._getValues(uint256).rTransferAmount (contracts/Token.sol#606) is too similar to Babycow._transferStandard(address,address,uint256).tTransferAmount (contracts/Token.sol#854)
Variable Babycow._transferToExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#870) is too similar to Babycow._transferStandard(address,address,uint256).tTransferAmount (cont
racts/Token.sol#854)
Variable Babycow._getValues(uint256).rTransferAmount (contracts/Token.sol#606) is too similar to Babycow._getTValues(uint256).tTransferAmount (contracts/Token.sol#615)
Variable Babycow._transferToExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#870) is too similar to Babycow._getTValues(uint256).tTransferAmount (contracts/Token.sol#615)
Variable Babycow._transferStandard(address,address,uint256).rTransferAmount (contracts/Token.sol#852) is too similar to Babycow._getTValues(uint256).tTransferAmount (contracts/Token.sol#615)
Variable Babycow._transferFromExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#889) is too similar to Babycow._transferBothExcluded(address,address,uint256).tTransferAmount
 (contracts/Token.sol#910)
Variable Babycow.reflectionFromToken(uint256,bool).rTransferAmount (contracts/Token.sol#549) is too similar to Babycow._transferBothExcluded(address,address,uint256).tTransferAmount (contracts/T
oken.sol#910)
Variable Babycow._transferFromExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#889) is too similar to Babycow._getValues(uint256).tTransferAmount (contracts/Token.sol#605)
Variable Babycow._getRValues(uint256,uint256,uint256,uint256,uint256).rTransferAmount (contracts/Token.sol#628) is too similar to Babycow._transferBothExcluded(address,address,uint256).tTransfer
Amount (contracts/Token.sol#910)
Variable Babycow.reflectionFromToken(uint256,bool).rTransferAmount (contracts/Token.sol#549) is too similar to Babycow._getValues(uint256).tTransferAmount (contracts/Token.sol#605)
Variable Babycow._getRValues(uint256,uint256,uint256,uint256,uint256).rTransferAmount (contracts/Token.sol#628) is too similar to Babycow._getValues(uint256).tTransferAmount (contracts/Token.sol
#605)
Variable Babycow._transferFromExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#889) is too similar to Babycow._transferToExcluded(address,address,uint256).tTransferAmount (
contracts/Token.sol#872)
Variable Babycow._getRValues(uint256,uint256,uint256,uint256,uint256).rTransferAmount (contracts/Token.sol#628) is too similar to Babycow._transferToExcluded(address,address,uint256).tTransferAm
ount (contracts/Token.sol#872)
Variable Babycow._transferFromExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#889) is too similar to Babycow._transferStandard(address,address,uint256).tTransferAmount (co
ntracts/Token.sol#854)
Variable Babycow._transferFromExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#889) is too similar to Babycow._getTValues(uint256).tTransferAmount (contracts/Token.sol#615)
Variable Babycow._getValues(uint256).rTransferAmount (contracts/Token.sol#606) is too similar to Babycow._transferBothExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#910)
Variable Babycow._transferToExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#870) is too similar to Babycow._transferBothExcluded(address,address,uint256).tTransferAmount (
contracts/Token.sol#910)
Variable Babycow.reflectionFromToken(uint256,bool).rTransferAmount (contracts/Token.sol#549) is too similar to Babycow._getTValues(uint256).tTransferAmount (contracts/Token.sol#615)
Variable Babycow._getRValues(uint256,uint256,uint256,uint256,uint256).rTransferAmount (contracts/Token.sol#628) is too similar to Babycow._transferStandard(address,address,uint256).tTransferAmou
nt (contracts/Token.sol#854)
Variable Babycow._getValues(uint256).rTransferAmount (contracts/Token.sol#606) is too similar to Babycow._getValues(uint256).tTransferAmount (contracts/Token.sol#605)
Variable Babycow._transferToExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#870) is too similar to Babycow._getValues(uint256).tTransferAmount (contracts/Token.sol#605)
Variable Babycow._getRValues(uint256,uint256,uint256,uint256,uint256).rTransferAmount (contracts/Token.sol#628) is too similar to Babycow._getTValues(uint256).tTransferAmount (contracts/Token.so
l#615)
Variable Babycow._transferStandard(address,address,uint256).rTransferAmount (contracts/Token.sol#852) is too similar to Babycow._transferBothExcluded(address,address,uint256).tTransferAmount (co
ntracts/Token.sol#910)
Variable Babycow._getRValues(uint256,uint256,uint256,uint256,uint256).rTransferAmount (contracts/Token.sol#628) is too similar to Babycow._transferFromExcluded(address,address,uint256).tTransfer
Amount (contracts/Token.sol#891)
Variable Babycow._transferStandard(address,address,uint256).rTransferAmount (contracts/Token.sol#852) is too similar to Babycow._getValues(uint256).tTransferAmount (contracts/Token.sol#605)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#variable-names-too-similar

Babycow.DEAD (contracts/Token.sol#405) should be constant
Babycow._decimals (contracts/Token.sol#380) should be constant
Babycow._name (contracts/Token.sol#378) should be constant
Babycow._symbol (contracts/Token.sol#379) should be constant
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-constant

Babycow._tTotal (contracts/Token.sol#383) should be immutable
Babycow.liquidityFeeonBuy (contracts/Token.sol#390) should be immutable
Babycow.liquidityFeeonSell (contracts/Token.sol#391) should be immutable
Babycow.marketingFeeonBuy (contracts/Token.sol#393) should be immutable
Babycow.marketingFeeonSell (contracts/Token.sol#394) should be immutable
Babycow.taxFeeonBuy (contracts/Token.sol#387) should be immutable
Babycow.taxFeeonSell (contracts/Token.sol#388) should be immutable
Babycow.totalBuyFees (contracts/Token.sol#400) should be immutable
Babycow.totalSellFees (contracts/Token.sol#401) should be immutable
Babycow.uniswapV2Pair (contracts/Token.sol#408) should be immutable
Babycow.uniswapV2Router (contracts/Token.sol#407) should be immutable
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-immutable
```

Result => A static analysis of contract's source code has been performed using slither,

No major issues were found in the output

FUNCTIONAL TESTING

Router (PCS V2):
0xD99D1c33F9fC3444f8101754aBC46c52416550D1

1- Adding liquidity **(passed):**
https://testnet.bscscan.com/tx/0x1c01076fcdea39ac5a79d06115bbe0c90d71a3026e0603ba8feb04a0a0d7d635

2- Buying when excluded from fees (0% tax) **(passed):**
https://testnet.bscscan.com/tx/0xd1b3fc1535bc4d362241f17d73c148e2cd70e7982bf9f72e436e873c24f40fda

3- Selling when excluded from fees (0% tax) **(passed):**
https://testnet.bscscan.com/tx/0xa2ed267d248734008909c7e9f1c55f490003e4aca02b823b460b6198413fb046

4- Transferring when excluded from fees (0% tax) **(passed):**
https://testnet.bscscan.com/tx/0xdc4e67715e3df7e8debb617fff1c59141082a502ce70612110904a005ebec1c5

5- Buying(0% tax) **(passed):**
https://testnet.bscscan.com/tx/0xff81691cb3eb97befb074e913abfd365d5a068ca0d46a699386aaec6e3433408

6- Selling (5% tax) **(passed):**
https://testnet.bscscan.com/tx/0xaa16e952662aec275c79d0627ad8ac92d8085ce0a059e43f2246b59e6369f827

FUNCTIONAL TESTING

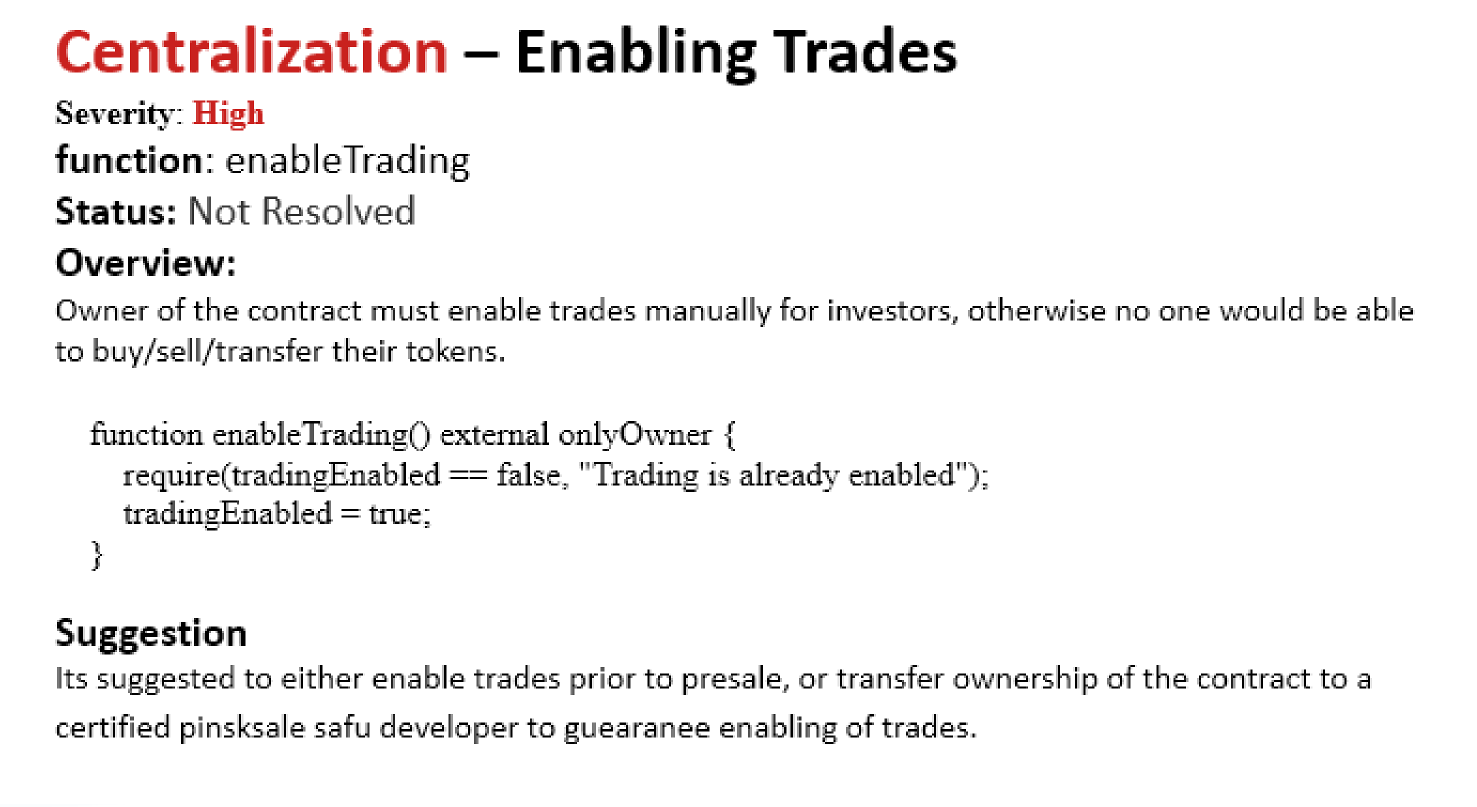
4- Transferring (0% tax) **(passed):**

https://testnet.bscscan.com/tx/0x23caa624064cbd9b54abf7db09ac440e2b1d0396df531494ed56e274557ff171

4- Internal swap **(passed):**

https://testnet.bscscan.com/address/0xbe09cabba1ddd2e7c3d48c771789e5d87c49b945#internaltx

FUNCTIONAL TESTING



Centralization – Enabling Trades

Severity: **High**

function: enableTrading

Status: Not Resolved

Overview:

Owner of the contract must enable trades manually for investors, otherwise no one would be able to buy/sell/transfer their tokens.

```
function enableTrading() external onlyOwner {
    require(tradingEnabled == false, "Trading is already enabled");
    tradingEnabled = true;
}
```

Suggestion

Its suggested to either enable trades prior to presale, or transfer ownership of the contract to a certified pinsksale safu developer to guearanee enabling of trades.

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 guarantees against the sale of team tokens or the removal of liquidity by the project audited in this document. Always Do your own research and protect yourselves from being scammed. The Auditace 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 Auditace receive a payment to manipulate those results or change the awarding badge that we will be adding in our website. Always 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 Auditace team disclaims any liability for the resulting losses.

ABOUT AUDITACE

We specializes in providing thorough and reliable audits for Web3 projects. With a team of experienced professionals, we use cutting-edge technology and rigorous methodologies to evaluate the security and integrity of blockchain systems. We are committed to helping our clients ensure the safety and transparency of their digital assets and transactions.

https://auditace.tech/

https://t.me/Audit_Ace

https://twitter.com/auditace_

https://github.com/Audit-Ace