RABBIDOGE

Smart Contract Audit Report

SYSFIXED

17 Feb 2023

TABLE OF CONTENTS

AUDITED DETAILS

Audited Project

Project name	Token ticker	Blockchain
RABBIDOGE	RDOGE	BSC

Addresses

Contract address	0x553a9A17cEf05Daa3a8Cc4e6df54B4fDF8925105
Contract deployer address	0x85958B0076E3Dc4007dF569bd554C10Af6BaA600

Project Website

https://www.rabbidoge.com/

Codebase

https://bscscan.com/address/0x553a9A17cEf05Daa3a8Cc4e6df54B4fDF8925105#code

SUMMARY

□□□□,□□□□ - Xīnnián kuàilè, Gōngx□ fācái. Rabbidoge, the #1 meme token for the Rabbit year. Benefit for the holders no unlocked tokens, audited, low Tax 3% for buy/sell (can not be changed), beautifully hand-drawn NFT giveaways daily, and a fun and exciting marketing approach. Come and join our community. We would be glad to have you with us. Let's have fun while we embark on our journey to the moon.

Contract Summary

Documentation Quality

RABBIDOGE provides a document with a very good standard of solidity base code.

- The technical description is provided clearly and structured and also don't have any risk issue.

Code Quality

The Overall quality of the basecode is GOOD

- Standart solidity basecode and rules are already followed with Coinhound Project .

Test Coverage

Test coverage of the project is 100% (Through Codebase)

Audit Findings Summary

- SWC-101 | Arithmetic operation Issues discovered on lines 445, 455, 463, 482, 484, 496, 497, 511, 513, 607, 608, 675, 697, 698, 713, 721, and 722.

CONCLUSION

We have audited the RABBIDOGE Coin which has released on January 2023 to discover issues and identify potential security vulnerabilities in RABBIDOGE Project. This process is used to find bugs, technical issues, and security loopholes that find some common issues in the code.

The security audit report produced satisfactory results with a low risk issue on the contract project.

The most common issue found in writing code on contracts that do not pose a big risk, writing on contracts is close to the standard of writing contracts in general. Some of the low issues that were found are only Arithmetic operation Issues.

AUDIT RESULT

Article	Category	Description	Result
Default Visibility	SWC-100 SWC-108	Functions and state variables visibility should be set explicitly. Visibility levels should be specified consciously.	PASS
Integer Overflow and Underflow	SWC-101	If unchecked math is used, all math operations should be safe from overflows and underflows.	ISSUE FOUND
Outdated Compiler Version	SWC-102	It is recommended to use a recent version of the Solidity compiler.	PASS
Floating Pragma	SWC-103	Contracts should be deployed with the same compiler version and flags that they have been tested thoroughly.	PASS
Unchecked Call Return Value	SWC-104	The return value of a message call should be checked.	PASS
SELFDESTRUCT Instruction	SWC-106	The contract should not be self-destructible while it has funds belonging to users.	PASS
Check-Effect Interaction	SWC-107	Check-Effect-Interaction pattern should be followed if the code performs ANY external call.	PASS
Assert Violation	SWC-110	Properly functioning code should never reach a failing assert statement.	PASS
Deprecated Solidity Functions	SWC-111	Deprecated built-in functions should never be used.	PASS
Delegate call to Untrusted Caller	SWC-112	Delegatecalls should only be allowed to trusted addresses.	PASS
DoS (Denial of Service)	SWC-113 SWC-128	Execution of the code should never be blocked by a specific contract state unless required.	PASS
Race Conditions	SWC-114	Race Conditions and Transactions Order Dependency should not be possible.	PASS

Authorization through tx.origin	SWC-115	tx.origin should not be used for authorization.	PASS
Block values as a proxy for time	SWC-116	Block numbers should not be used for time calculations.	PASS
Signature Unique Id	SWC-117 SWC-121 SWC-122	Signed messages should always have a unique id. A transaction hash should not be used as a unique id.	PASS
Shadowing State Variable	SWC-119	State variables should not be shadowed.	PASS
Weak Sources of Randomness	SWC-120	Random values should never be generated from Chain Attributes or be predictable.	PASS
Incorrect Inheritance Order	SWC-125	When inheriting multiple contracts, especially if they have identical functions, a developer should carefully specify inheritance in the correct order. The rule of thumb is to inherit contracts from more /general/ to more /specific/.	PASS

SMART CONTRACT ANALYSIS

Started	Mon Jan 16 2023 08:43:55 GMT+0000 (Coordinated Universal Time)
Finished	Tue Jan 17 2023 09:43:55 GMT+0000 (Coordinated Universal Time)
Mode	Standard
Main Source File	RDOGE.sol

Detected Issues

ID	Title	Severity	Status
SWC-101	ARITHMETIC OPERATION "-" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "+" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "-" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "-" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "+=" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "+=" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "+=" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "-" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "-=" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "*" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "/" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "+" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "/" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "-" DISCOVERED	low	acknowledged

SWC-101	ARITHMETIC OPERATION "/" DISCOVERED	low	acknowledged
SWC-110	OUT OF BOUNDS ARRAY ACCESS	low	acknowledged
SWC-110	OUT OF BOUNDS ARRAY ACCESS	low	acknowledged

SWC-101 | ARITHMETIC OPERATION "-" DISCOVERED

LINE 445

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- RDOGE.sol

Locations

```
444   unchecked {
445   _approve(sender, _msgSender(), currentAllowance - amount);
446   }
447   }
448
```

SWC-101 | ARITHMETIC OPERATION "+" DISCOVERED

LINE 455

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- RDOGE.sol

Locations

```
454   function increaseAllowance(address spender, uint256 addedValue) public virtual
returns (bool) {
455   _approve(_msgSender(), spender, _allowances[_msgSender()][spender] + addedValue);
456   return true;
457   }
458
```

SWC-101 | ARITHMETIC OPERATION "-" DISCOVERED

LINE 463

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- RDOGE.sol

Locations

```
462   unchecked {
463   _approve(_msgSender(), spender, currentAllowance - subtractedValue);
464   }
465
466   return true;
```

SWC-101 | ARITHMETIC OPERATION "-" DISCOVERED

LINE 482

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- RDOGE.sol

Locations

```
481   unchecked {
482   _balances[sender] = senderBalance - amount;
483   }
484   _balances[recipient] += amount;
485
```

SWC-101 | ARITHMETIC OPERATION "+=" DISCOVERED

LINE 484

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- RDOGE.sol

Locations

```
483   }
484   _balances[recipient] += amount;
485
486   emit Transfer(sender, recipient, amount);
487
```

SWC-101 | ARITHMETIC OPERATION "+=" DISCOVERED

LINE 496

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- RDOGE.sol

Locations

```
495
496   _totalSupply += amount;
497   _balances[account] += amount;
498   emit Transfer(address(0), account, amount);
499
```

SWC-101 | ARITHMETIC OPERATION "+=" DISCOVERED

LINE 497

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- RDOGE.sol

Locations

```
496   _totalSupply += amount;
497   _balances[account] += amount;
498   emit Transfer(address(0), account, amount);
499
500   _afterTokenTransfer(address(0), account, amount);
```

SWC-101 | ARITHMETIC OPERATION "-" DISCOVERED

LINE 511

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- RDOGE.sol

Locations

```
510   unchecked {
511   _balances[account] = accountBalance - amount;
512   }
513   _totalSupply -= amount;
514
```

SWC-101 | ARITHMETIC OPERATION "-=" DISCOVERED

LINE 513

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- RDOGE.sol

Locations

```
512   }
513   _totalSupply -= amount;
514
515   emit Transfer(account, address(0), amount);
516
```

SWC-101 | ARITHMETIC OPERATION "*" DISCOVERED

LINE 607

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- RDOGE.sol

Locations

```
606
607   _mint(owner(), 1e9 * (10 ** decimals()));
608   swapTokensAtAmount = totalSupply() / 500_000;
609
610   tradingEnabled = false;
```

SWC-101 | ARITHMETIC OPERATION "/" DISCOVERED

LINE 608

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- RDOGE.sol

Locations

```
607   _mint(owner(), 1e9 * (10 ** decimals()));
608   swapTokensAtAmount = totalSupply() / 500_000;
609
610   tradingEnabled = false;
611   swapEnabled = false;
```

SWC-101 | ARITHMETIC OPERATION "+" DISCOVERED

LINE 675

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- RDOGE.sol

Locations

```
674   to == uniswapV2Pair &&
675   marketingFeeOnBuy + marketingFeeOnSell > 0 &&
676   swapEnabled
677   ) {
678   swapping = true;
```

SWC-101 | ARITHMETIC OPERATION "/" DISCOVERED

LINE 697

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- RDOGE.sol

Locations

```
696   if (_totalFees > 0) {
697   uint256 fees = (amount * _totalFees) / 100;
698   amount = amount - fees;
699   super._transfer(from, address(this), fees);
700   }
```

SWC-101 | ARITHMETIC OPERATION "-" DISCOVERED

LINE 698

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- RDOGE.sol

Locations

```
697   uint256 fees = (amount * _totalFees) / 100;
698   amount = amount - fees;
699   super._transfer(from, address(this), fees);
700   }
701
```

SWC-101 | ARITHMETIC OPERATION "/" DISCOVERED

LINE 713

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- RDOGE.sol

Locations

```
712   function setSwapTokensAtAmount(uint256 newAmount) external onlyOwner{
713   require(newAmount > totalSupply() / 1_000_000, "SwapTokensAtAmount must be greater
than 0.0001% of total supply");
714   swapTokensAtAmount = newAmount;
715
716   emit SwapTokensAtAmountUpdated(swapTokensAtAmount);
```

SWC-110 | OUT OF BOUNDS ARRAY ACCESS

LINE 721

low SEVERITY

The index access expression can cause an exception in case of use of invalid array index value.

Source File

- RDOGE.sol

Locations

```
720   address[] memory path = new address[](2);
721   path[0] = address(this);
722   path[1] = address(USDT);
723
724   uniswapV2Router.swapExactTokensForTokens(
```

SWC-110 | OUT OF BOUNDS ARRAY ACCESS

LINE 722

low SEVERITY

The index access expression can cause an exception in case of use of invalid array index value.

Source File

- RDOGE.sol

Locations

```
721   path[0] = address(this);
722   path[1] = address(USDT);
723
724   uniswapV2Router.swapExactTokensForTokens(
725   tokenAmount,
```

DISCLAIMER

This report is subject to the terms and conditions (including without limitation, description of services, confidentiality, disclaimer and limitation of liability) set forth in the Services Agreement, or the scope of services, and terms and conditions provided to you ("Customer" or the "Company") in connection with the Agreement. This report provided in connection with the Services set forth in the Agreement shall be used by the Company only to the extent permitted under the terms and conditions set forth in the Agreement. This report may not be transmitted, disclosed, referred to, or relied upon by any person for any purposes, nor may copies be delivered to any other person other than the Company, without Sysfixed's prior written consent in each instance.

This report is not, nor should be considered, an "endorsement" or "disapproval" of any particular project or team. This report is not, nor should be considered, an indication of the economics or value of any "product" or "asset" created by any team or project that contracts Sysfixed to perform a security assessment. This report does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors, business, business model, or legal compliance.

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 us on the basis of what it says or doesn't say, or how we produced it, and it is important for you to conduct your own independent investigations before making any decisions. We go into more detail on this in the below disclaimer below – please make sure to read it in full.

This report should not be used in any way to make decisions around investment or involvement with any particular project. This report in no way provides investment advice, nor should be leveraged as investment advice of any sort. This report represents an extensive assessing process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology.

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 Sysfixed and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers, and other representatives) (Sysfixed) owe no duty of care.

ABOUT US

Sysfixed is a blockchain security certification organization established in 2021 with the objective to provide smart contract security services and verify their correctness in blockchain-based protocols. Sysfixed automatically scans for security vulnerabilities in Ethereum and other EVM-based blockchain smart contracts. Sysfixed a comprehensive range of analysis techniques—including static analysis, dynamic analysis, and symbolic execution—can accurately detect security vulnerabilities to provide an in-depth analysis report. With a vibrant ecosystem of world-class integration partners that amplify developer productivity, Sysfixed can be utilized in all phases of your project's lifecycle. Our team of security experts is dedicated to the research and improvement of our tools and techniques used to fortify your code.